



Escola d'Enginyeria de Telecomunicació i
Aeroespacial de Castelldefels

UNIVERSITAT POLITÈCNICA DE CATALUNYA

TREBALL DE FI DE CARRERA

TÍTOL DEL TFC : Estudi, dimensionament i impacte del sistema de guiatge del vehicle autònom

TÍTOL: Enginyeria de Sistemes de Telecomunicació

AUTOR: Mario Jiménez Feijóo

DIRECTOR: Dr. Antoni Elias Fusté

DATA: 20 de juny de 2017

Title : Study, sizing and impact of the autonomous vehicle's guiding system

Author: Mario Jiménez Feijóo

Director: Dr. Antoni Elias Fusté

Date: 20 de juny de 2017

Overview

Throughout the recent history of cities internal combustion engine vehicles have been and still are the mainstream mean of transport. As a result of this, some of their inherent problems such as social and environmental impact and even mortality have always been present in our society and dealt with as an unavoidable lesser evil.

Nowadays, current scientific and technologic development allows us to finally face the problem by evolving traffic systems with two goals in mind: to increase their efficiency and to remove their past downsides.

This project introduces a new traffic guiding model designed for urban areas. In this model there are only autonomous vehicles which are driven by an artificial intelligence instead of classic ones. These new vehicles are routed through 'beacons' deployed throughout the city which act as routers or navigation processing nodes. When a vehicle approach a beacon its IA receives new instructions to reach the next beacon in the path to the destination. This process repeats until the vehicle reaches the destination.

The proposed model borrows some ideas from computer networking principles. Thus, a vehicle could be considered as equivalent to an IP packet. Since a fundamental working piece of IP networks are dynamic routing protocols, it is clear that our model will need an equivalent system to work as well.

Despite of our efforts to mimetize computer networks, vehicle traffic networks have some differences (some of them are physically inherent) and therefore some limitations arise. Therefore the same dynamic routing protocols which work for IP networks won't work out of the box for vehicle networks, thus requiring being completely redefined for this new environment.

Since redifining and implementing a dynamic routing protocol in order to make it fully compatible with vehicle networks would be a very harsh task, in order to better reproduce these particular conditions which current routing protocols are not designed to take into account, we introduce a Python application which help us analyze this problematic.

Aquest projecte final de carrera està dedicat a:

El meu director de projecte Antoni Elias Fusté per donar-me l'oportunitat de treballar en aquest fascinant projecte, el qual m'ha permès aplicar alguns conceptes apresos durant el grau, així com conèixer una infinitat de nous.

Tots els professors de l'EETAC que s'han implicat en transmetre'm inquietuds, esperit d'auto-superació i passió pel que he estat fent durant aquests anys. Ha estat un plaer aprendre d'ells.

Al meu germà Sergio, i l'Alex perquè sempre ha estat al meu costat quan he necessitat ànim, suport i comprensió i perquè la seva ajuda amb aquest treball ha estat inestimable.

Els meus pares Sílvia y Jose, per tot el que han fet per mi per que això fos possible i perquè sempre m'han recolzat.

A la Cristina i el Josep Marc, per totes les bones estones compartides durant tantes classes i treballs en grup al llarg d'aquests anys, el seu recolzament anímic ha estat vital.

ÍNDEX

INTRODUCCIÓ	1
CAPÍTOL 1. Proposta de model de sistema de guiatge	3
1.1. Plantejament de l'escenari a implementar	3
1.2. Protocols d'encaminament dinàmic	6
1.2.1. Elecció del protocol d'encaminament	7
1.3. Subnetting	9
1.4. Exercici d'anàlisi de l'escenari	11
1.5. Implementació amb software GNS3	12
1.5.1. Configuració dels routers	13
1.5.2. Sumarització	14
1.5.3. Mètrica d'EIGRP	18
1.5.4. Afectació del delay	18
1.5.5. Afectació del bandwidth	22
1.6. Limitacions del sistema de guiatge	23
1.6.1. Limitacions per IPv4	23
1.6.2. Limitacions per EIGRP	24
CAPÍTOL 2. Millores del sistema de guiatge	25
2.1. Algoritme de cerca A*	25
2.2. Explicació del procediment i les funcions a implementar	27
2.2.1. Escenari a implementar	29
2.2.2. Representació matricial de l'escenari	29
2.2.3. Cerca del node més proper	30
2.2.4. Heurística	30
2.2.5. Cost	31
2.2.6. Heurística global	32
2.2.7. Expansió d'un node	32
2.2.8. Eliminar cicles	32
2.2.9. Eliminar camins redundants	32
2.2.10. Inserció ordenada	34

2.3. Anàlisi de routes	34
2.3.1. Ruta A-I	36
2.3.2. Ruta G-R	37
2.3.3. Ruta O-F	39
2.3.4. Comparació amb Dijkstra	41
 CAPÍTOL 3. Conclusions i línies de futur	 43
3.1. Conclusions generals	43
3.1.1. Impacte social	45
3.1.2. Línies futures	46
 BIBLIOGRAFIA	 49
 APÈNDIX A. Història i evolució del trànsit vehicular	 51
A.1. Aparició de l'automòbil i conseqüències vials	51
A.1.1. Història de l'automòbil	51
A.1.2. Naixement de la senyalització transitària	52
A.1.3. Inicis del semàfor	54
A.1.4. Nova urbanització	55
A.2. Impacte medioambiental	57
A.2.1. Partícules en suspensió	57
A.2.2. Gasos nocius	58
A.2.3. Altres contaminants	58
A.2.4. Conseqüències en la qualitat de l'aire	59
A.2.5. Mortalitat per contaminació atmosfèrica	61
A.3. Mortalitat en vies de trànsit	61
A.4. Ineficiències del trànsit i possibles solucions	63
A.4.1. Acceleració no simultània	63
A.4.2. Traffic snake	64
A.4.3. Platooning	65
A.4.4. La conducció humana	66
A.5. El vehicle autònom	66
A.5.1. Avantatges del vehicle autònom	67
A.5.2. Funcionament del vehicle autònom	68
 APÈNDIX B. Protocols d'encaminament dinàmic en detall	 69
B.0.1. RIP: Routing Information Protocol	69

B.0.2. OSPF: Open Shortest Path First	70
B.0.3. IS-IS: Intermediate System to Intermediate System	71
B.0.4. BGP: Border Gateway Protocol	73
B.0.5. EIGRP: Enhanced Interior Gateway Routing Protocol	74
APÈNDIX C. Subnetting en detall	77
C.0.1. Subdivisió per barri	77
C.0.2. Subdivisió per zona	78
C.0.3. Subdivisions per carrer	81
APÈNDIX D. Taules de rutes	83
APÈNDIX E. Funcions i fitxers de l'aplicació Astar	95

INTRODUCCIÓ

Ens trobem en una societat en la qual el trànsit vehicular no ha patit canvis realment revolucionaris des de fa més de 100 anys. Seguim tenint carreteres amb tot tipus d'indicacions estàtiques i vehicles amb motors del segle XXI controlats per la ineficient voluntat humana que encara es mouen a través d'explosions internes de combustibles fòssils.

Aquest projecte planteja una visió revolucionària i canviant respecte la situació present; imaginem una ciutat a la qual fem desaparèixer tota senyal de trànsit a excepció dels passos de vianants, eliminem per complet els vehicles de motor combustió reemplaçant-los per vehicles autònoms.

Els vehicles autònoms actualment disposen d'una gran intel·ligència artificial que els permet arribar al seu destí sense requerir d'intervenció humana, tot i que finalment s'adapten a les condicions de trànsit actuals convivint amb la resta de vehicles no intel·ligents. L'objectiu d'aquest projecte és dissenyar un sistema de guiatge que, comptant amb tota la informació del trànsit, pugui triar en tot moment la ruta més adequada per cada vehicle en funció de l'estat actual de les vies i el destí.

Aquest sistema distribuït és homòleg al concepte de xarxa de commutació de paquets; sent l'equivalent a un paquet, un vehicle. Seguint la mateixa lògica, equiparant els enllaços a carrers i els nodes a les interseccions on les quals trobem una balisa que actua com a element enrutador. El resultat final és una arquitectura sòlida des de la qual podem accedir a la informació qualsevol node autoritzat així com propagar la informació a la resta de nodes de manera ràpida i segura. És a dir, una topologia escalable on qualsevol balisa és capaç de dirigir tot vehicle al seu destí de la manera més òptima possible, donant lloc a un sistema distribuït.

Donat el concepte de sistema distribuït, necessitem definir un procés d'intercanvi d'informació. Aquesta eina es coneix a les xarxes de commutació de paquets com a protocol d'encaminament dinàmic. Els principals protocols d'encaminament dinàmic són: RIP, OSPF, BGP, EIGRP i el desconegut IS-IS. Un dels objectius principals consisteix en conèixer els avantatges i inconvenients que presenta cada protocol per tal de poder prendre'ls com a punt de partida per finalment poder desenvolupar un nou model de trànsit vehicular.

Aquest model basat en entorn xarxa presentarà unes certes limitacions les quals faran que el sistema no s'acabi d'adaptar a l'entorn del trànsit, motiu pel qual es proposaran solucions programades en un petit escenari en llenguatge Python que implementarà entre d'altres, l'algoritme de cerca A*.

Per últim, s'analitzarà des d'un punt de vista social i medioambiental l'adopció d'aquest sistema, així com idees de futur sobre un possible modelatge de la ciutat del futur.

CAPÍTOL 1. PROPOSTA DE MODEL DE SISTEMA DE GUIATGE

El transport automobilístic ha suposat una gran revolució, esdevenint actualment un medi de transport vital en l'humanitat. Tot i tenir moltes bondats de les quals ens hem estat aprofitant fins ara, també presenta una sèrie de repercussions urbanístiques, socials, medioambientals i mortals (més informació a l'apèndix A). Es tracta d'un medi de transport el qual no ha patit grans canvis en els darrers anys. Seguim desplaçant-nos amb motors que funcionen a base d'explosions d'un combustible limitat i perjudicial.

A més a més, resulta prou ineficient degut a la lenta (i sovint perillosa) resposta dels humans davant la conducció. Aquest projecte es centra en el nou concepte de vehicle automatitzat, i planteja treure'n el màxim potencial d'aquest, donat que actualment es tracta d'un concepte inicialment destinat a interactuar amb la resta de vehicles no-automatitzats, és a dir, detecta, respecta i obeeix els senyals vials, comportant-se com un vehicle més.

Per tant, des d'aquest punt de vista no estem aprofitant tot el seu potencial donat que estariem adaptant aquests conceptes a les infraestructures actuals (infraestructures que es van fer en el seu dia en conseqüència de l'arribada de l'automòbil) en lloc de remodelar el sistema de trànsit actual. El propòsit d'aquest projecte és el de reinventar el model de trànsit que coneixem actualment, centrant-nos concretament en dissenyar un nou sistema de guiatge del que en coneixerem més detalls a continuació.

1.1. Plantejament de l'escenari a implementar

Estant situats a la ciutat de Barcelona com a entorn en els que ens basarem a mode d'exemple; imaginem una ciutat a la qual eliminem per complet els vehicles de motor de combustió reemplaçant-los per vehicles autònoms, i per últim, fem desaparèixer tota senyal de trànsit a excepció dels passos de vianants o semàfors (només per a vianants).

Amb aquest dràstic canvi, passariem a tenir una ciutat amb uns nivells de contaminació tant atmosfèrics com acústics mínims en quant a activitat de transport automobilístic. A més a més, no serien necessaris els senyals de trànsit actuals, eliminant per tant tot tipus de manteniment necessari i evitant accidents en situacions senyalitzades de manera ambigua o incorrecta. A conseqüència d'aquest agressiu canvi, la principal dificultat és la coordinació i el guiament dels vehicles sense la senyalització actual.

Els vehicles autònoms actualment disposen d'una gran intel·lència artificial que els permet arribar al seu destí sense requerir d'intervenció humana significativa (la persona al volant només ha d'estar pendent per qüestions de seguretat), tot i que finalment s'adapten a les condicions de trànsit actuals convivint amb la resta de vehicles no intel·ligents.

Per tal d'aprofitar tot el potencial tecnològic del que disposen els vehicles autònoms, proposem la invenció d'un nou sistema de guiatge que, comptant amb tota la informació del trànsit, triï la ruta més adient per a cada vehicle en funció de l'estat actual de les vies i el destí, avolint congestions, retencions i garantitzant una certa seguretat.

Aquesta visió és homòloga al concepte de xarxa de commutació de paquets; sent l'equivalent d'un paquet, un vehicle. Seguint la mateixa lògica, equiparem els enllaços a carrers i els nodes a les interseccions, on trobem una balisa que actua com a element enrutador de vehicles.

Cal comentar que el disseny d'aquest sistema de guiatge no prescindeix de la navegació per GPS, que en tot moment és capaç d'indicar la posició i la ruta a seguir en cas de fallada del sistema, ni tampoc de la visió per computador dels vehicles autònoms, que els permeten analitzar l'entorn i actuar en conseqüència (controlar distàncies, frenar quan sigui necessari, etc).

El resultat final és una arquitectura sòlida des de la qual podem accedir a la informació de qualsevol node autoritzat així com propagar la informació a la resta de nodes de manera ràpida i segura.

És a dir, una topologia escalable on qualsevol balisa és capaç de dirigir qualsevol vehicle al seu destí de la manera més òptima possible, donant lloc a un sistema distribuït.



Figura 1.1: Mapejat del districte de l'Eixample de BCN amb equivalent en entorn xarxa

Els principals motius pels quals optem per implementar una arquitectura basada en xarxes de commutació de paquets IP a l'urbanisme automobilístic són els següents:

- **Monitorització del trànsit:** amb el disseny basat en entorn xarxa, seria possible conèixer la informació de qualsevol node, és a dir, podríem conèixer la càrrega momentània de qualsevol carrer (enllaç), i calcular les possibles rutes en funció d'aquests paràmetres.
- **Alta escalabilitat:** resulta viable i lògic afegir més nodes a la xarxa així com interconnectar diferents zones (altres sistemes distribuïts).

- **Ràpida convergència:** en el moment en què s'efectua un canvi en algun node, la idea és que aquest es propagui ràpidament a la resta de nodes. Per exemple, si necessitem canviar el límit de velocitat d'un enllaç o bé tallar un carrer, aquest últim podria consistir en deixar d'anunciar una ruta determinada a la resta de nodes veïns.
- **Prioritat als serveis d'emergència:** en relació a l'anterior punt, seria possible prioritzar el trànsit d'ambulàncies, bombers o policia entre d'altres donat que si considerem l'homologia abans esmentada entre vehicle-paquet, cada vehicle podria tenir el seu camp identificatiu (i per tant, prioritzar-ne els necessaris) de la mateixa manera que entenem el concepte de *capçalera* o *header* de paquets IP.
- **Habilitació del Platooning:** gràcies a la intel·ligència artificial dels vehicles autònoms, tots els vehicles executarien la marxa alhora, a més a més, estarien constantment comunicats entre ells per tal de tenir controlada la situació. Un altre factor a tenir en compte és que els vehicles que comparteixen part o totalitat del seu destí, poden viatjar junts com si es tractés d'un *tren de carretera*, que és precisament aquest concepte.
- **Eliminació de la direcció prohibida:** la direcció prohibida existeix per dos principals motius; un és degut a la simple limitació física dels carrers que només disposen de l'amplada suficient per a la circulació d'un vehicle. L'altre motiu és degut a la dificultat logística que presentaria imposar una configuració de doble sentit en tots els carrers, donat que s'haurien d'adoptar regles de trànsit dinàmiques que resultarien inaplicables pels humans.

La direcció prohibida no té sentit respecte l'equivalència que plantegem ja que els enllaços d'una xarxa només tenen sentit si són bidireccionals, un carrer amb direcció prohibida equivaldria a un cost idealment infinit. Però l'objectiu no és una adaptació total del model actual, sino una millora revolucionària de l'actual sistema de trànsit.

Per tant, plantegem un sistema de trànsit bidireccional amb l'objectiu d'assolir en tot moment la ruta més òptima sense saturar les artèries principals de la ciutat. Degut a què no es planteja en cap moment cedir el control del vehicle a humans, no resultaria costós circular sense senyalització si considerem una interacció contínua amb la resta de vehicles.



Figura 1.2: Exemple de futur vehicle biplaça

Per últim, plantegem abolir la limitació física de l'amplada dels carrers mitjançant vehicles biplaça com el de la Figura 1.2, en els quals la disposició dels seients és davantera i posterior en lloc de places contigües com estem acostumats. D'aquesta manera seria possible aprofitar la infraestructura de vies limitades a un únic sentit i convertir-les en vies bidireccionals.

Un cop plantejat l'escenari, és necessari definir un procés que s'encarregarà d'intercanviar la informació al conjunt de xarxes per tal de que cada node sigui capaç de determinar el conjunt de rutes que farà servir per enviar els paquets. Estem parlant per tant dels *protocols d'encaminament dinàmic*.

1.2. Protocols d'encaminament dinàmic

La informació presentada en aquesta secció ha estat extreta de les fonts [1], [2] i [3].

Un protocol d'encaminament dinàmic es defineix com un conjunt de processos, algorismes i missatges que s'utilitzen amb el fi d'intercanviar informació d'encaminament. Aquest intercanvi d'informació es realitza utilitzant taules d'enrutament que resumeixen l'elecció dels millors camins que realitza el protocol per tal de poder dirigir o enrotar els paquets cap a les diferents xarxes.

Hi ha protocols d'encaminament estàtics i dinàmics.

- **Protocols d'Encaminament Estàtic:** són generats pel propi administrador, totes les rutes estàtiques que se li ingressin són les que el router coneixerà, per tant podrà encaminar paquets cap a aquestes xarxes.
- **Protocols d'Encaminament Dinàmic:** amb un protocol d'enrutament dinàmic, l'administrador només s'encarrega de configurar el protocol d'enrutament mitjançant ordres 'IOS' en tots els routers de la xarxa, i aquests, automàticament intercanviaran les seves taules d'encaminament amb els seus routers veïns. Aquest precissament és el comportament desitjat del nostre sistema.

Abans de parlar sobre la classificació dels protocols d'enrutament dinàmics, és necessari de parlar d'un concepte anomenat *Mètrica*. La mètrica és l'anàlisi, i en el que es basa l'algoritme del protocol d'enrutament dinàmic per triar i preferir una ruta per sobre d'una altra. Basant-se en això, el protocol crearà la taula d'enrutament en el router, publicant només les millors rutes.

Els protocols d'enrutament dinàmics es classifiquen en:

- **Vector Distància:** la seva mètrica es basa en el que s'anomena en xarxes '*Nombre de Salts*', és a dir, la quantitat de nodes (routers) pels quals ha de passar el paquet per arribar a la xarxa destí, sent la ruta òptima aquella que tingui el menor nombre de salts.

- **Estat d'Enllaç:** la seva mètrica es basa en el retard, amplada de banda, càrrega i fiabilitat dels diferents enllaços possibles per arribar a una destinació. En base a aquests paràmetres, el protocol prefereix una ruta per sobre d'una altra. Aquests protocols utilitzen un tipus de publicacions anomenades *Publicacions d'estat d'enllaç* (LSA) que s'intercanvien entre els routers. Mitjançant aquestes publicacions cada router crea una base dades de la topologia de la xarxa completa.

1.2.1. Elecció del protocol d'encaminament

De la informació sintetitzada principalment a partir de les fonts [2] i [3], obtenim la següent taula comparativa dels protocols d'encaminament dinàmic. A l'**apèndix B** trobareu un resum en detall de cada un dels protocols RIP, OSPF, IS-IS, BGP i EIGRP, junt amb un raonament sobre si són bons candidats per a una possible extrapolació trànsit-xarxa.

Comparativa de protocols d'encaminament					
Protocol	RIP	OSPF	IS-IS	BGP	EIGRP
Tipus	Dist-Vector	Link state	Link state	Path-Vect	Adv.Dist-Vector
Convergència	Lenta	Ràpida	Ràpida	Lenta	Molt ràpida
Cost computacional	Molt baix	Alt	Alt	Baix	Baix
Escalament	Molt baix	Normal	Normal	Molt alt	Alt
Lliure/Propietari	Lliure	Lliure	Lliure	Lliure	Propietari

En quant a la comparativa anterior, descartem en primera instància RIP i BGP.

El protocol RIP és excessivament simple, la seva convergència és lenta, i el fet principal pel qual no s'adapta a les nostres necessitats és el seu límit de 15 salts, sent 16 salts un destí inassolible. Per tant no és gens escalable, sent l'escalabilitat un dels factors clau en un desplegament de les característiques que plantegem en aquest projecte.

BGP és un protocol que no té gran relació amb la resta donat que es basa en polítiques i no en l'estat del canal a l'hora d'elaborar rutes. Defineix el concepte de *Sistema autònom* i dona preferència a rutes les quals hagin de recórrer un menor nombre de SA's, independentment d'altres factors. Tot i que és molt escalable (de fet és el protocol que utilitza internet), compta amb una convergència molt lenta, pel que no és adequat per aquest projecte.

En quant a la resta de protocols: OSPF, IS-IS i EIGRP, distingim ràpidament el tipus, ja que OSPF i IS-IS es basen en 'Link state' i EIGRP en 'Advanced distance vector', el qual és un vector distància que compta amb moltes característiques del tipus 'Link state'. OSPF i IS-IS són dos protocols molt semblants on la principal diferència rau en la seva jerarquització; sent a través d'àrees a OSPF, i a través de nivells a IS-IS.

Ambdós protocols disposen d'una bona convergència, a més a més de ser prou escalables, tot i que compten amb un alt cost computacional degut a què tot router coneix el mapejat de la xarxa. A conseqüència d'aquest fet, augmentar els nodes suposa un increment de memòria junt amb un major ús de CPU en els càlculs de Dijkstra.

La major problemàtica que suposen IS-IS i OSPF és la seva jerarquització, ja que tal i com aquest projecte està plantejat (un desplegament de balises per tota la ciutat de Barcelona), observant la Figura 1.3 suposaria el següent:



Figura 1.3: Exemple de desplegament d'àrees amb OSPF a Barcelona

A mode d'exemple senzill, si per exemple vulguéssim anar de Pedralbes (Àrea 1) a Sarrià (Àrea 2), amb aquest disseny el protocol ens encaminaria de l'àrea 1 a la backbone 0, i finalment a l'àrea 2. Aquest recorregut no seria gens eficient degut a la impossibilitat d'interconnectar entre si àrees diferents a la 0. En el cas d'IS-IS seria un cas semblant però plantejat en nivells.

Si bé IS-IS és més flexible donat que permet fer redistribucions de rutes, és difícil de configurar i ràpidament es generen bucles d'encaminament si no es controla quines rutes es propaguen (i a quins nodes).

EIGRP en canvi és un protocol molt més versàtil ja que disposa d'una convergència molt ràpida, un consum de recursos moderat i un escalament molt bo. Aquest protocol no planteja jerarquitzacions per àrea o nivell ni utilitza *Dijkstra* per determinar el *Shortest Path*.

Utilitza una mètrica composta per tal de determinar la millor ruta, basant-se en l'amplada de banda, el retard, la confiabilitat i la càrrega de l'enllaç, conceptes que són molt extrapolables al món del trànsit vehicular. El fet que cada router conegui la informació a través dels seus veïns (*routing by rumour*) no fa necessari una gran quantitat de memòria tot i no triar exactament les millors rutes degut a què no incorpora Dijkstra.

Aquest concepte és molt diferent al del 'Link state' donat que en comptes d'estar continuament transmetent informació dels enllaços, només es transmeten quan s'efectua un canvi. El seu funcionament basat en *Routing by rumour* aconsegueix que es tracti d'un protocol molt lleuger, a més a més de fiable donat que l'algoritme '*Diffusing Update Algorithm*' (DUAL) que implementa, permet la inserció de rutes en cas de falla de la ruta principal; executant el canvi a la ruta de resguard de manera immediata.

És per aquests motius que utilitzarem **EIGRP** per a l'elaboració del sistema de guiatge.

1.3. Subnetting

La informació presentada en aquesta secció ha estat extreta de les fonts [4], [5], [6] i [7].

Un cop triat el protocol d'encaminament dinàmic, és hora de construir un escenari a mode il·lustratiu que ens permeti contemplar les característiques de basar el trànsit vehicular en entorn xarxa com hem comentat fins el moment.

Interpretarem cada intersecció com un node (router) i cada segment entre nodes com un conjunt de carrers (hosts) que pertanyen a una mateixa xarxa entre aquests dos nodes. Per tant, cada segment estarà comprés en diferents xarxes respectivament, esdevenint tota adreça IP única en tot l'escenari.

Partirem d'una única xarxa de classe A amb l'adreça següent: **10.0.0.0/8**. Aquesta xarxa compta amb un total de $2^{32-8} - 1$ adreça de xarxa - 1 adreça broadcast = 6,777,214 hosts.

Una adreça IP és un adreçament usat per a identificar únicament un dispositiu en una xarxa IP. L'adreçament es compon de 32 bits binaris, que poden ser divisibles en una porció de la xarxa i la identifiquen amb l'ajuda d'una *màscara de subxarxa*. Els 32 bits binaris es divideixen en quatre octets. Una màscara de xarxa ajuda a detectar quina part de la direcció identifica la xarxa i quina part de la direcció identifica el node.

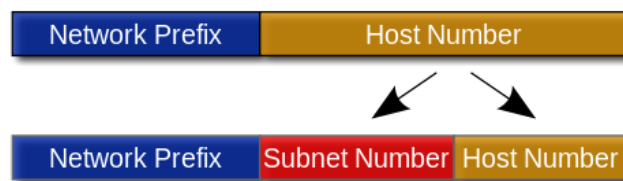


Figura 1.4: Identificador de xarxa i host number en relació al subnetting [4]

Considerant la ciutat de Barcelona a mode d'exemple pràctic en aquest projecte, tindrem en compte els següents aspectes:

- Barcelona compta amb 10 barris reconeguts, sent necessaris **4 bits** per tal de representar-los.
- Cada barri es subdivideix en zones, sent Horta Guinardó el barri que té més zones (10) i per tant aquest serà el nostre llinar mínim. Per tal de representar les zones necessitariem uns altres **4 bits** addicionals.
- Cada zona es subdivideix en un conjunt de carrers, per tant suposarem un ús de **8 bits**. És a dir, 256 possibles números de carrers entre un node i el següent,.

L'objectiu és subdividir la xarxa 10.0.0.0/8 en subxarxes menors (i a la seva vegada, subdividir-les de nou els cops necessaris) de manera que puguem representar una xarxa independent per cada barri de Barcelona. En resum, una subxarxa per cada barri de Barcelona, una subxarxa per cada zona que tingui un barri i per últim, un cojunt de subxarxes per cada zona que representin els carrers.

Aquest plantejament considera la necessitat de poder distingir en tot moment a quina zona pertany un determinat carrer a partir de la seva direcció IP, així com una zona d'un barri i viceversa. D'aquí neix el concepte de la necessitat de la màscara de subxarxa, ja que mitjançant una multiplicació bit a bit (AND) entre la màscara de subxarxa i una direcció de host particular, podem obtenir la direcció identificadora de la xarxa.

Aleshores, representar una divisió per barris suposaria una ampliació de la màscara de subxarxa en 4 bits addicionals (màscara de /8 a /12), així com 4 bits addicionals per representar les zones (màscara de /12 a /16) i per últim, 8 bits per representar els carrers, acabant finalment amb una màscara /24.

Per tal de dividir aquesta gran xarxa en subxarxes lògiques menors, farem l'anomenat procés de *Subnetting*. L'objectiu és que cadascuna d'aquestes subxarxes treballi tant a nivell d'enviament com de recepció de paquets com una xarxa individual, encara que totes pertanyin a la mateixa xarxa física i al mateix domini.

El procés de Subnetting permet una millor administració, control del trànsit i seguretat en segmentar la xarxa per funció. També millora el rendiment de la xarxa en reduir el trànsit de broadcast de la nostra xarxa.

Com desavantatge, la seva implementació malgasta moltes direccions donat que per cada subxarxa que creem, necessitem com a mínim 2 adreces, 1 per l'adreça de xarxa i 1 per l'adreça de broadcast. A la Figura 1.5 podem observar un esquema del plantejament del Subnetting d'aquest projecte:

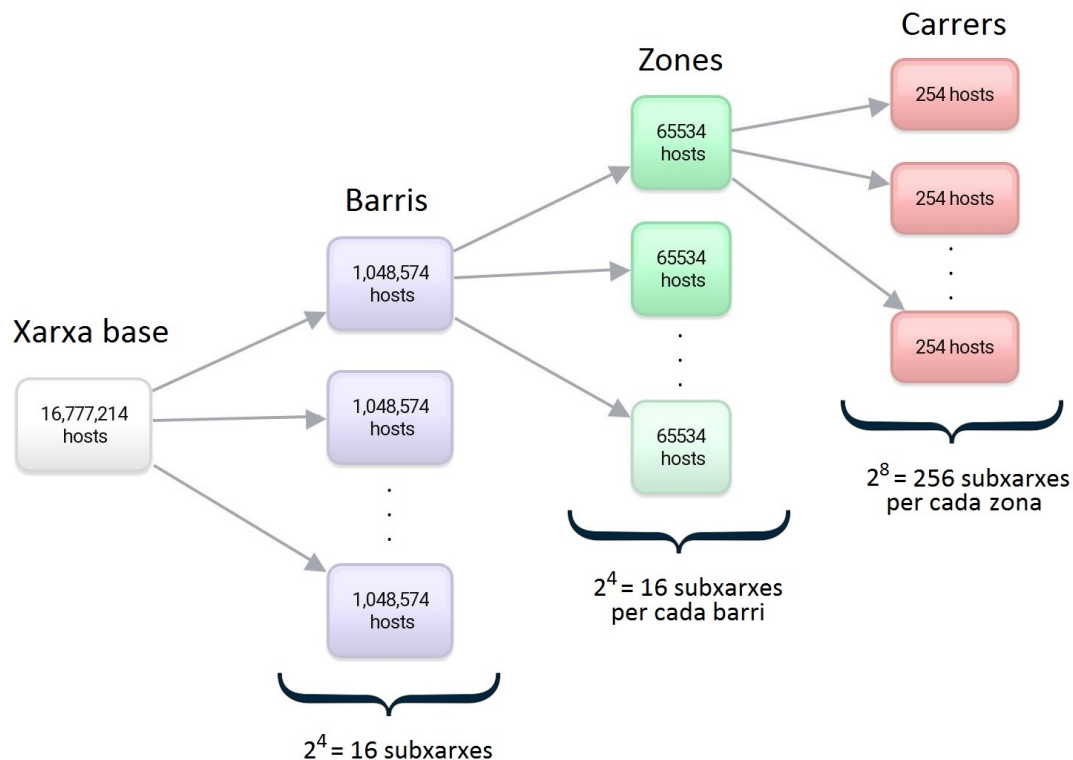


Figura 1.5: Esquema del plantejament del Subnetting

1.4. Exercici d'anàlisi de l'escenari

Un cop realitzat el Subnetting (tot el procés consta en detall a l'**apèndix C**), construïm el següent escenari:

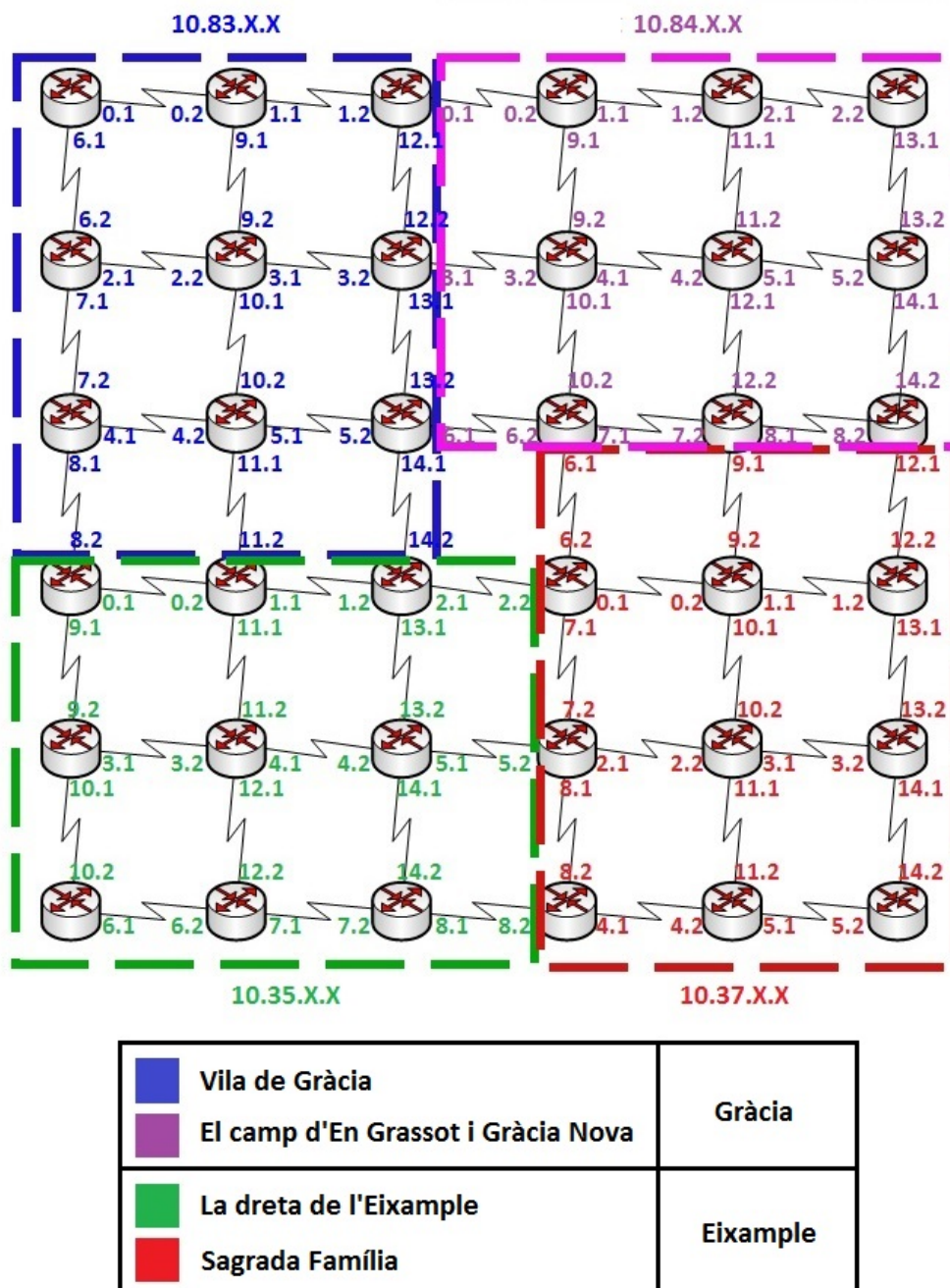


Figura 1.6: Escenari a implementar

Com podem observar, es tracta d'un escenari amb 36 routers que podrien representar un cas real simplificat. L'escenari compta amb dos barris coneguts, els quals compten cada un amb dues respectives zones.

L'objectiu a partir d'ara serà el de simular aquest escenari amb una eina adequada que

ens permeti observar el seu comportament sota el protocol EIGRP, junt amb les limitacions que comporta implementar aquest sistema en una xarxa de trànsit vehicular.

1.5. Implementació amb software GNS3

El software *GNS3* [8] és un simulador gràfic d'entorn xarxa que permet la simulació de xarxes complexes. Proporciona simulacions completes i precises gràcies a la seva vinculació amb *Dynamips* [9], un emulador d'IOS de Cisco, i *Qemu*, un emulador de sistemes operatius de lliure origen.

Per últim, també integra *VirtualBox*, un programari de virtualització gratuït i de gran abast.

És a dir, ens permet configurar unitats virtuals de Routers, Switches i Hosts (entre altres) així com observar el comportament de l'enviament i recepció d'informació, ja sigui en una petita xarxa LAN o bé un escenari tan extens com és l'internet.

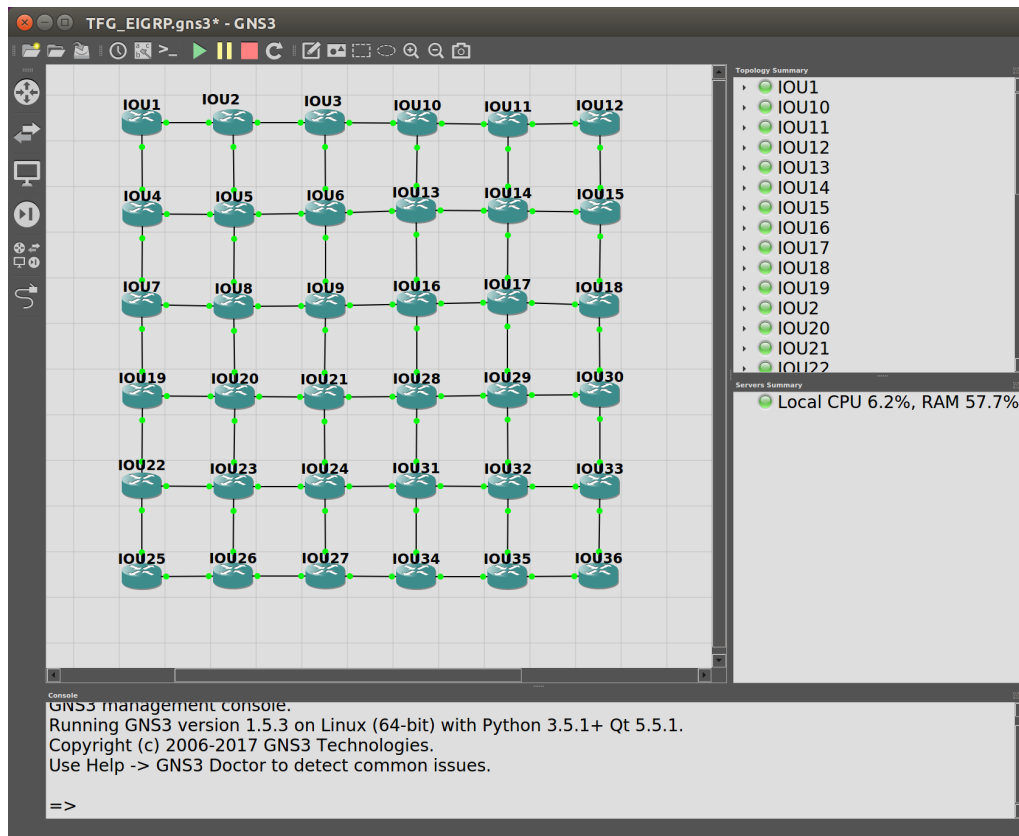


Figura 1.7: Escenari de xarxes implementat en GNS3

Els manuals de la font [9] han resultat de gran ajuda a l'hora de configurar el software, així com aprendre el seu funcionament.

1.5.1. Configuració dels routers

Configurarem cada router de l'escenari segons la planificació de la Figura 1.6.

```
hostname IOU2
!
interface Ethernet0/0
 ip address 10.83.0.2 255.255.255.0
 no shutdown
!
interface Ethernet0/2
 ip address 10.83.1.1 255.255.255.0
 no shutdown
!
interface Ethernet0/3
 ip address 10.83.9.1 255.255.255.0
 no shutdown
!

router eigrp TFG
!
 address-family ipv4 unicast autonomous-system 1
!
  topology base
  exit-af-topology
  network 10.0.0.0
 exit-address-family
```

Com podem observar, la informació llistada correspon a un exemple de l'assignació de direccions IP del router IOU2 a les respectives interfícies de sortida, junt amb la configuració del protocol EIGRP a cada router. Cada un dels 36 routers s'haurà de configurar amb els paràmetres adients.

Un cop configurades i estant en funcionament totes les interfícies de xarxa necessàries, exigirem que el router IOU1 ens mostri la seva taula d'encaminament EIGRP. Les figures de l'annex D.1, D.2 i D.3 corresponen a la taula de rutes de només el router IOU1. Com podem observar, tenim pràcticament 100 entrades amb un escenari de només 36 routers, que resulta una quantitat ínfima al costat d'un desplegament a nivell de ciutat.

La immediata preocupació d'aquest fet rau en què en cas d'aplicar aquest projecte a gran escala en tota la ciutat de Barcelona, estem parlant de milers de balises distribuïdes, és a dir, milers de nodes amb les seves corresponents taules de rutes. Aquest fet es resumiria en un augment gradual i consecutiu dels registres conforme s'augmenten els nodes, i per tant, sent necessari disposar de grans quantitats de memòria a cada node.

Tal i com s'ha plantejat l'escenari fins ara, no seria realitzable a gran escala. Per tal de solucionar aquesta problemàtica rau el concepte de **sumarització** [10].

1.5.2. Sumarització

Els routers poden arribar a gestionar taules d'enrutament tan grans (de vegades diversos milers o centenars de milers de rutes) que la complexitat de l'administració de les matrixes, sumat a la càrrega i el consum de recursos (memòria, CPU, amplada de banda, entre d'altres) siguin elements molt importants amb els quals els enginyers de xarxes han de bregar per tal de lliurar un servei eficient, segur i fiable en els seus sistemes.

Una de les tècniques utilitzades per optimitzar els recursos en aquest tipus de situacions és la sumarització o creació de superxarxes, també denominat *Supernetting*.

La creació de xarxes sumaritzades permet reduir considerablement les entrades a la taula d'enrutament en resumir la informació d'adreçament de dues o més subxarxes en un sol bloc IP.

En el cas de l'exercici d'aquest projecte, com sabem està format per dos barris (2 subxarxes principals) on cada barri té dues zones (2 subxarxes a cada subxarxa principal). Una manera de reduir les taules d'encaminament seria fent que els routers que actuen de frontera entre un barri i un altre s'anunciïn respectivament les seves direccions de xarxa genèriques.

Ens ajudarem de la font [10] per tal de configurar una sumarització de rutes sota el protocol EIGRP.

1.5.2.1. Sumarització per barri

El barri de Gràcia està representat per la xarxa: **10.80.0.0/12**

El barri de l'Eixample està representat per la xarxa: **10.32.0.0/12**

Aleshores, si un vehicle que es troba al barri de Gràcia (independentment de la zona) per exemple a la direcció 10.83.5.3 i es vol dirigir a la direcció 10.37.4.5, farem la multiplicació binària (AND) de la màscara de subxarxa del seu barri: 255.224.0.0 pel destí 10.37.4.5, obtindrem l'identificador de xarxa de l'Eixample: 10.32.0.0.

Un cop trobat l'identificador de xarxa, el que sabem és que volem assolir un destí fora de la nostra subxarxa de barri ja que l'operació ens ha retornat un identificador de xarxa diferent al del barri d'inici de la trajectòria.

Fins ara hem obtingut una gran taula d'encaminament donat que disposàvem de rutes per arribar fins a qualsevol router final. Ara en canvi, el que es proposa és que cada cop que ens vulguem dirigir a un altre barri, les balises o nodes del nostre barri només sàpiguen dirigir-nos fins la balisa de l'altre barri destí més propera, i un cop ja estant en la subxarxa del barri destí, les balises interiors sabrien dirigir-nos al nostre destí.

Si bé és cert que sacrificaríem eficiència donat que ja no tindrem els camins més òptims en el moment de dirigir-nos a un altre barri, almenys conseguiríem reduir la taula de rutes considerablement.

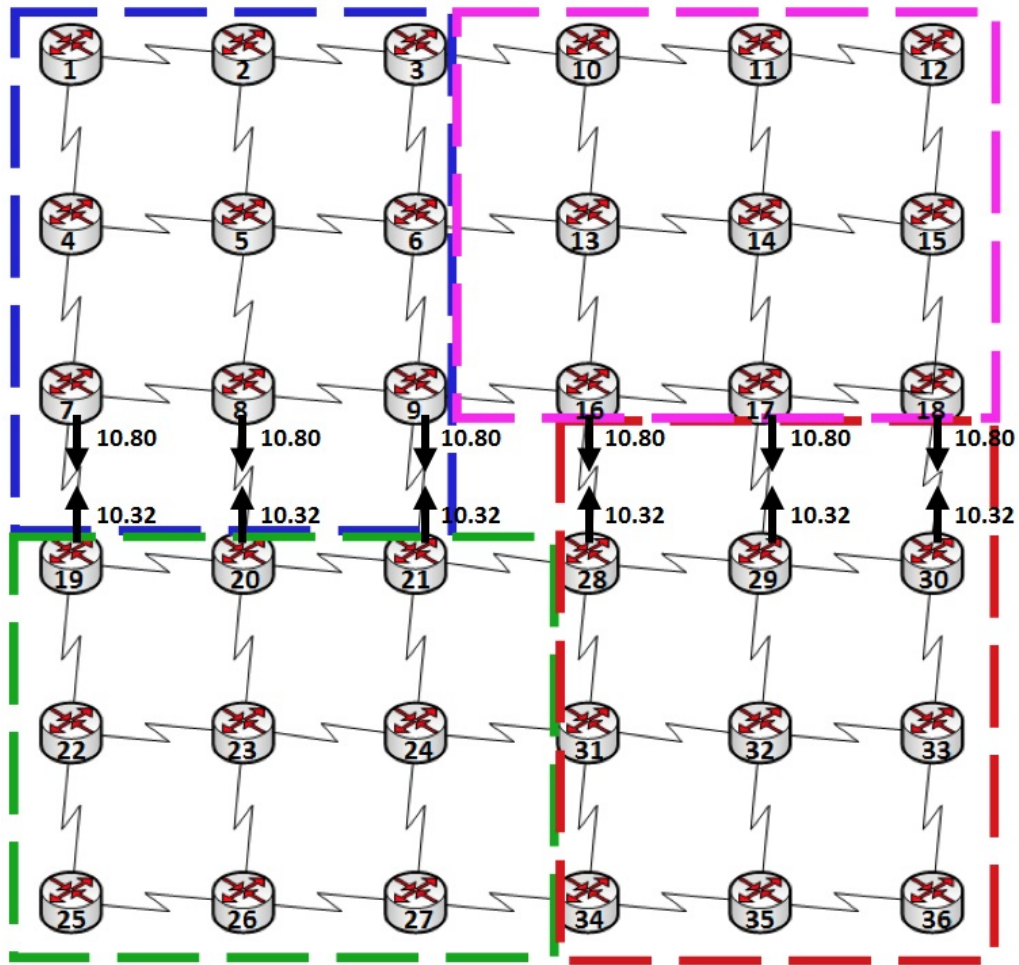


Figura 1.8: Pla de sumarització per barri

Per tal d'aconseguir aquest resultat, els routers 7, 8, 9, 16, 17 i 18 pertanyents al barri de Gràcia anunciaran per les seves interfícies Eth0/3 (interfícies 'inferiors' segons el mapa) la seva adreça 10.80.0.0/12.

```
router eigrp TFG
!
address-family ipv4 unicast autonomous-system 1
  af-interface Eth0/3
    summary-address 10.80.0.0 255.240.0.0
```

Així com els routers 19, 20, 21, 28, 29 i 30 pertanyents al barri de l'Eixample, anunciaran per les seves interfícies Eth0/1 (interfícies 'superiors' segons el mapa) la seva adreça 10.32.0.0/12.

```
router eigrp TFG
!
address-family ipv4 unicast autonomous-system 1
  af-interface Eth0/1
    summary-address 10.32.0.0 255.240.0.0
```

D'aquesta manera, per tal d'accedir a un barri diferent, cada router només tindrà una única adreça per sortir a l'exterior de la subxarxa.

Cal dir que per tal d'enrutar per l'interior, seguirem tenint tant les rutes per adjacència directa com qualsevol altra ruta dintre de la subxarxa de barri.

A l'annex, a la Figura D.4 i D.5 trobem la nova taula d'encaminament del router 1 (pertanyent al barri de Gràcia), i per últim a la Figura D.6 i D.7 trobem la taula d'encaminament del router 25 (ja que és el router en situació a oposada a R1 a l'esquema).

1.5.2.2. Sumarització total

Un cop havent sumaritzat per barri, extenem el concepte a les zones. Per tant, si dintre d'un barri ens volem desplaçar a un altre, només en sabrem arribar fins al router més proper de la nostra zona que ens permeti arribar a l'altra zona, i un cop estant a la zona de destí, les balises interiors del barri ens encaminarien amb exactitud.

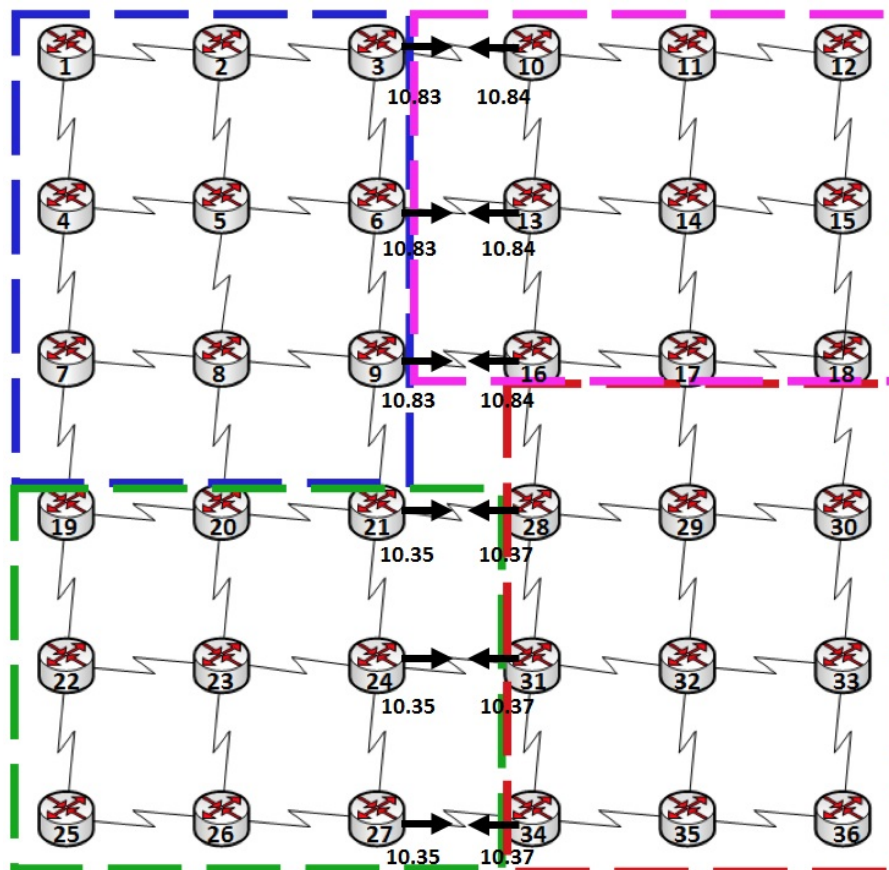


Figura 1.9: Pla de sumarització per zona

Aquesta nova sumarització no exclueix l'anterior, sino que es combina de manera que en el cas més extrem, podrem suposar el següent exemple:

Estem al barri de l'Eixample, a la zona de 'La dreta de l'Eixample' concretament amb la direcció 10.35.10.4 i ens volem desplaçar fins al barri de Gràcia, concretament a la zona de 'El camp d'En Grassot i Gràcia nova' a la direcció 10.84.1.5.

El primer que faria el router de partida és la multiplicació binària (AND) de la màscara de subxarxa del seu barri: 255.224.0.0 pel destí 10.84.1.5, obtenint aleshores l'identificador de xarxa de Gràcia: 10.80.0.0. Aleshores el router encaminarà el vehicle fins al router de l'altre barri més proper ja que ha denotat que la direcció destí no pertany al barri.

Un cop al barri de Gràcia, probablement el vehicle estigui per la zona de 'Vila de Gràcia'. Aleshores es faria el mateix procediment que abans: multiplicaríem bit a bit la màscara de subxarxa de la zona de 'Vila de Gràcia' que correspon a 255.255.0.0 per l'adreça destí 10.84.1.5, i aquesta operació ens atorgaria la direcció de xarxa 10.84.0.0 corresponent a 'El camp d'En Grassot i Gràcia nova'.

Al notar que aquesta direcció és exterior a la subxarxa de zona, les balises encaminarien al vehicle fins al node més proper a la zona destí corresponent, i al arribar a aquesta, es faria l'encaminament fins a la direcció de host final.

Per tal d'aconseguir aquest resultat, els routers **3,6 i 9** pertanyents a la zona de 'Vila de Gràcia' anunciaran per les seves interfícies Eth0/2 (interfícies dretes segons el mapa) la seva adreça 10.83.0.0/16.

```
router eigrp TFG
!
address-family ipv4 unicast autonomous-system 1
  af-interface Eth0/2
    summary-address 10.83.0.0 255.255.0.0
```

Faríem servir una comanda similar amb:

- Els routers **10, 13 i 16** pertanyents a la zona de 'El camp d'En Grassot i Gràcia nova', anunciaran per les seves interfícies Eth0/0 (interfícies 'esquerres' segons Fig. 1.9) la seva adreça 10.84.0.0/16.
- Els routers **21, 24 i 27** pertanyents a la zona de 'La dreta de l'Eixample', anunciaran per les seves interfícies Eth0/2 (interfícies 'dretes' segons Fig. 1.9) la seva adreça 10.37.0.0/16.
- Els routers **28, 31 i 34** pertanyents a la zona de 'Sagrada Família', anunciaran per les seves interfícies Eth0/0 (interfícies 'esquerres' segons Fig. 1.9) la seva adreça 10.37.0.0/16.

Un cop finalitzades les sumaryzacions, cada router hauria de tenir una taula amb totes les rutes per arribar a qualsevol node de la seva zona, i només algunes rutes singulars per arribar a diferents zones i barris, reduint considerablement el contingut de les taules respecte de les situacions anteriors.

A la Figura D.8 trobarem la taula d'encaminament del router 1, així com a la Figura D.9 la taula d'encaminament del router 12, Figura D.10 pel router 25, i per últim, Figura D.11 pel router 36. Noteu que es tracta dels 4 routers situats als extrems de la topologia.

Com es pot observar, les noves taules d'encaminament compten amb unes 25 rutes a diferència de les pràcticament 100 rutes que teníem a cada router abans de sumaritzar barri i zona, pel que es pot afirmar que és un procés vital a l'hora d'optimitzar la memòria dels routers.

1.5.3. Mètrica d'EIGRP

Sabem que EIGRP compta amb una mètrica composta a través d'una fórmula que computa l'amplada de banda, el retard, la confiabilitat i la càrrega de l'enllaç [3], però fins a quin punt s'adapta al projecte?

Per defecte, un router amb EIGRP compta amb la següent configuració:

```

Routing Protocol is "eigrp 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Default networks flagged in outgoing updates
  Default networks accepted from incoming updates
  EIGRP-IPv4 VR(TFG) Address-Family Protocol for AS(1)
    Metric weight K1=1, K2=0, K3=1, K4=0, K5=0 K6=0
    Metric rib-scale 128
    Metric version 64bit
    Soft SIA disabled
    NSF-aware route hold timer is 240
    Router-ID: 10.37.14.2
  
```

Com podem observar, per defecte només K_1 i K_3 s'estableix a 1, i la resta a zero. Basant-se en la simple fórmula: $(\text{Bandwidth} + \text{Delay}) \cdot 256$.

1.5.4. Afectació del delay

El *delay* o *retard*, és un valor acumulatiu (mesurat en desenes de microsegons) que respon a la suma dels valors de delay de cada segment del camí (path) [3].

Aquest paràmetre és informat de router a router de manera acumulativa. Per tant, si modifiquem el delay d'un determinat router a una interfície d'entrada, el router veí rebrà aquest valor i afegirà el seu propi valor de delay.

Per defecte el delay ve fixat en 1000us:

```

IOU21#show interfaces Eth0/2 | i DLY
  MTU 1500 bytes , BW 10000 Kbit/sec , DLY 1000 usec ,
  
```


L'objectiu serà analitzar fins a quin punt és útil modificar el delay en funció del comportament que desitgem. Una manera d'analitzar el seu comportament seria esbrinar el camí o traça de nodes consecutius pels quals viatja un paquet fins a arribar a l'adreça destí. Un cop sabuda la traça, observarem en què es basa per prendre una decisió davant de camins equivalents i la influència del delay en aquest aspecte.

Per tal d'analitzar-lo plantejarem l'escenari següent:

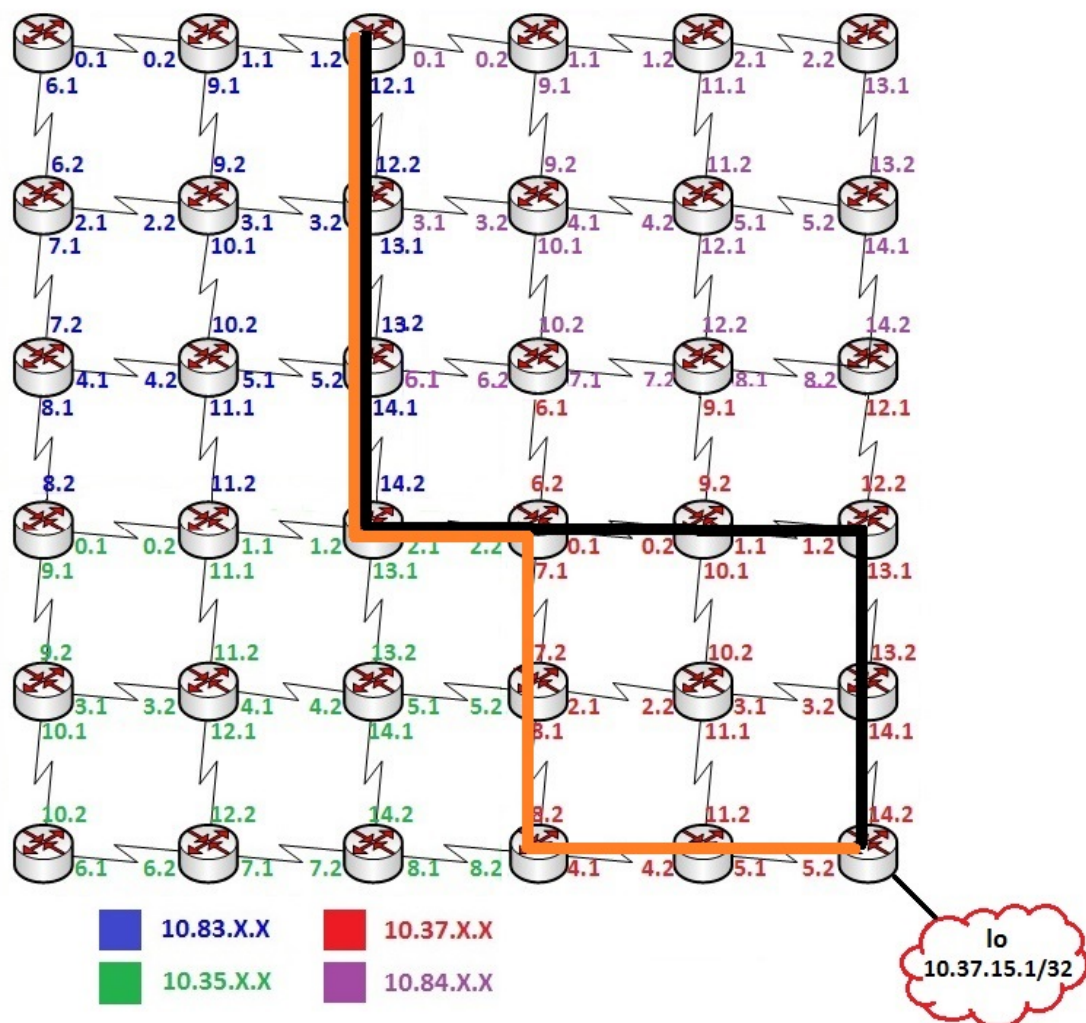


Figura 1.10: Escenari amb loopback creat

A la Figura 1.10 tenim l'escenari de sempre però amb la inclusió d'un *loopback*, (una interfície de xarxa virtual).

```
IOU36(config)#interface lo0
IOU36(config-if)#ip address 10.37.15.1 255.255.255.255
```

El motiu d'haver-la creat és degut a què si volem esbrinar quina trajectòria es dona des de l'adreça 10.83.2.1 fins al router 36, hem d'indicar una direcció concreta, pel que si indiquéssim com a destí la 10.37.5.2 podríem estar forçant que el paquet entrés per la interfície Eth0/0 del router 36.

Per tant, indicant una adreça virtual que penja d'una tercera interfície del router 36, podrem observar com es comporta en casos d'igualtat.

A continuació utilitzarem la comanda *traceroute* per tal d'esbrinar la trajectòria que segueix el paquet des de l'adreça 10.83.12.1 fins la loopback 10.37.15.1.

```
IOU3>traceroute 10.37.15.1
Tracing the route to 10.37.15.1
VRF info: (vrf in name/id, vrf out name/id)
 1 10.83.12.2 0 msec 0 msec 0 msec
 2 10.83.13.2 0 msec 1 msec 0 msec
 3 10.83.14.2 0 msec 1 msec 0 msec
 4 10.35.2.2 0 msec 1 msec 0 msec
 5 10.37.7.2 1 msec 1 msec 0 msec
 6 10.37.8.2 1 msec 1 msec 0 msec
 7 10.37.4.2 1 msec 2 msec 1 msec
 8 10.37.5.2 1 msec 1 msec *
```

Aquesta trajectòria correspon a la marcada de **taronja** a la Figura 1.10. Per què ha pres aquest camí i per exemple no ha fet un desviament seguint el camí negre (mateix cost)?

```
IOU28#show ip route 10.37.15.1
Routing entry for 10.37.15.1/32
  Known via "eigrp 1", distance 90, metric 2560640, type internal
  Redistributing via eigrp 1
  Last update from 10.37.7.2 on Ethernet0/3, 00:16:20 ago
  Routing Descriptor Blocks:
    10.37.7.2, from 10.37.7.2, 00:16:20 ago, via Ethernet0/3
      Route metric is 2560640, traffic share count is 1
      Total delay is 4002 microseconds
      Reliability 255/255, minimum bandwidth is 10000 Kbit
      Loading 1/255, Hops 4
    * 10.37.0.2, from 10.37.0.2, 00:16:20 ago, via Ethernet0/2
      Route metric is 2560640, traffic share count is 1
      Total delay is 4002 microseconds
      Reliability 255/255, minimum bandwidth is 10000 Kbit
      Loading 1/255, Hops 4
```

Aquí podem apreciar com en el router 28 (el de la bifurcació) compta amb dues possibles rutes amb el mateix delay (4002 microsegons).

La resposta la té el mecanisme '**CEF load balancing**' que incorpora EIGRP.

Segons Cisco, *CEF load balancing* [3] és un terme que descriu una funcionalitat en un router que distribueix paquets a través de múltiples enllaços sobre la base de capa 3 d'informació d'enrutament.

Bàsicament té la funció de repartir el trànsit entre diferents rutes per tal de controlar la càrrega del sistema. Amb el CEF load balancing d'EIGRP amb costos desiguals, s'assignarà un nombre de *buckets* inversament proporcionals a la relació mètrica.

Per tant, si la mètrica d'una ruta és el doble que l'altre, obtindrà la meitat de buckets que l'altre camí.

Els buckets suposen una certa probabilitat, és a dir, quants més buckets tenen una ruta, més probable és que el paquet sigui encaminat per aquest camí. En el nostre escenari les dues rutes tenien la mateixa probabilitat.

Si extraïem la informació d'aquest mecanisme al GNS3, obtenim el següent:

```
IOU28#show ip cef 10.37.15.1 internal
10.37.15.1/32, epoch 0, RIB[1], refcnt 5, per-destination sharing
sources: RIB
feature space:
  IPRM: 0x00028000
ifnums:
  Ethernet0/2(5): 10.37.0.2
  Ethernet0/3(6): 10.37.7.2
path list F5F0D574, 11 locks, per-destination, flags 0x49
  path F5E36014, share 0/1, type attached nexthop, for IPv4
    nexthop 10.37.0.2 Ethernet0/2, IP adj out of Ethernet0/2
  path F5E3607C, share 1/1, type attached nexthop, for IPv4
    nexthop 10.37.7.2 Ethernet0/3, IP adj out of Ethernet0/3
output chain:
  loadinfo F3A7AB6C, per-session, 2 choices, flags 0083, 9 locks
  flags [Per-session, for-rx-IPv4, 2buckets]
  2 hash buckets
    < 0 > IP adj out of Ethernet0/2, addr 10.37.0.2 F44D09C0
    < 1 > IP adj out of Ethernet0/3, addr 10.37.7.2 F44D0890
```

Com podem observar, cada ruta té 1 bucket i aleshores es troben en condicions iguals. Al trobar-se en mateixes condicions, es calcula un hash en funció de l'adreça IP origen i IP destí, que en aquest cas resulta en el hash < 1 >. D'aquesta manera, amb alguna altra possible adreça en igualtat de condicions, el trànsit podria no ser encaminat pel mateix camí, conseguint aleshores equilibrar el trànsit.

Per últim, si modifiquem el delay a la interfície Eth0/2 del router 21:

```
IOU21#show interfaces Eth0/2 | i DLY
  MTU 1500 bytes, BW 10000 Kbit/sec, DLY 1000 usec,

IOU21(config)#interface Eth0/2
IOU21(config-if)#delay 10000

IOU21#show interfaces Eth0/2 | i DLY
  MTU 1500 bytes, BW 10000 Kbit/sec, DLY 100000 usec,
```

Al executar la comanda *traceroute* de nou, observem la seva traça modificada:

```
IOU3#traceroute 10.37.15.1
Type escape sequence to abort.
Tracing the route to 10.37.15.1
VRF info: (vrf in name/id, vrf out name/id)
 1 10.83.12.2 0 msec 0 msec 0 msec
 2 10.83.13.2 1 msec 0 msec 0 msec
 3 10.83.14.2 0 msec 1 msec 0 msec
 4 10.35.13.2 0 msec 0 msec 0 msec
 5 10.35.5.2 0 msec 1 msec 0 msec
 6 10.37.8.2 1 msec 0 msec 0 msec
 7 10.37.4.2 1 msec 1 msec 0 msec
 8 10.37.5.2 1 msec * 2 msec
```

1.5.5. Afectació del bandwidth

La informació exposada en aquesta subsecció ha estat sintetitzada de les fonts [3] i [11].

A diferència de la majoria de les mètriques, l'amplada de banda mínima correspon a l'amplada de banda mínim de tot el recorregut, pel que no reflecteix la quantitat de nodes pels que passem amb connexions de baixa velocitat en el trajecte, només es tindrà en compte la pitjor.

Alterar el paràmetre del bandwidth per tal de forçar a EIGRP per fer servir un determinat path no és una bona idea:

- El canvi només afectarà al path només en el cas en què el bandwidth introduït sigui el menor de tots, això té com a principal motiu evitar els colls d'ampolla.
- EIGRP per defecte està pensat per utilitzar el 50% del bandwidth. Disminuir aquest valor podria causar que mai rebessin paquets certs veïns d'EIGRP per falta de recursos. Així com configurar un bandwidth elevat per sobre de les capacitats físiques portarà a EIGRP a consumir més amplada de banda de l'existent, amb el consegüent problema de pèrdua de paquets.

En canvi, modificar el delay no suposa un impacte negatiu, és simplement una constant acumulativa que, segons Cisco, és l'adequada per tal de condicionar la selecció del best-path.

1.6. Limitacions del sistema de guiatge

El sistema de guiatge presenta comportaments interessants en el món del trànsit com ara els següents:

- És **confiable**, EIGRP compta amb rutes de resguard en cas de fallada de rutes principals.
- **Altament escalable**, permet formar una gran topologia de nodes, sobretot si es sumaritza.
- **Convergència ràpida**: quan s'efectua un canvi, es propaga de manera ràpida a la resta de nodes per difusió, sent possible propagar nova informació d'encaminament que deixi de tenir en compte l'anterior.
- **Aprofitament d'infraestructura urbana**: a nivell urbanístic no és necessari invertir força en canviar infraestructures vials. Només tractaria sobre un desplegament de routers per tota la ciutat, podent fins i tot aprofitar infraestructura de xarxa ja instal·lada com per exemple fibra òptica ja desplegada per la ciutat.

Però trobem certes mancances tant a nivell d'IP com del propi protocol d'encaminament (EIGRP en aquest cas) les quals fan que si bé podria ser aplicable fins a cert punt, no acaba d'estar completament adaptat.

1.6.1. Limitacions per IPv4

- **Time To Live (TTL)**: és un concepte utilitzat per indicar per quants nodes pot *saltar* un paquet com a màxim fins a assolir el seu destí. El valor màxim del TTL en IP és de 255. Aquest valor va decreixent conforme travessa nodes, de manera que si aquest valor es veu reduït a 0, el paquet es descarta.

Aquest concepte no té sentit en el trànsit vehicular ja que és possible travessar més de 255 nodes en arribar a un determinat destí, sobretot en cas d'extendre el sistema de guiatge en tot el país, o fins i tot, entre països.

Si bé aquest paràmetre té sentit en xarxa degut a què en un moment determinat es pot donar un bucle o un canvi de topologia, un vehicle no es pot 'descartar' ni tampoc es pot trobar en un bucle en un moment determinat. Per tant seria un aspecte molt controlat que difereix del plantejat en IP.

- **Exhauriment d'adreces d'IPv4** [12]: El creixement exponencial d'Internet està portant cap a l'esgotament de les adreces IPv4. Aquest tema ha estat una preocupació des dels anys 80. Com a conseqüència, s'ha convertit en el factor impulsor en la creació i adopció de diverses noves tecnologies com *NAT* i posteriorment *IPv6*.

Si el sistema de guiatge es vulgués aplicar sota IPv4 com s'ha plantejat en el projecte, i a més a més es vulgués aplicar a una escala considerable, aquest presentaria certa limitació donat que en IPv4 una direcció IP es representa amb 32 bits, podent proporcionar 2^{32} direccions úniques, el qual resulta insuficient.

No obstant això, IPv6 admet 2^{128} adreces úniques. Una adreça IP en aquesta versió està composta per vuit segments de 2 bytes cadascun, sumant un total de 128 bits. Una manera de solucionar aquesta limitació seria aleshores implementant IPv6.

1.6.2. Limitacions per EIGRP

- **Problema *Stuck in active*** [13]: el concepte fa referència a que el router està involucrat en un càlcul de la difusió d'un nou camí a alguna xarxa, i aquest càlcul sembla estar estancat a causa de que algunes respostes esperades no han arribat en un temps raonable.

En EIGRP el procés per localitzar una ruta de substitució a una xarxa per a la qual s'ha perdut el camí utilitzable consisteix en realitzar sol·licituds als routers veïns. Si la consulta ha afectat al veí que la rep de tal manera que també va perdre la ruta (que podria haver passat si aquest router ens utilitzava com el seu next-hop), enviarà una consulta als seus propis veïns, emplenant la xarxa de consultes cícliques i consumint recursos.

Si en aquesta cadena d'esdeveniments, qualsevol router té problemes per contestar a les consultes, la difusió de rutes no podrà ser completada, de manera que la nova ruta cap a la destinació no podrà ser establerta. És el problema greu que té EIGRP, i el qual es pot evitar treballant amb 'àrees' petites i sumariant per reduir la latència.

- **Rutes no-òptimes:** EIGRP pren decisions sobre decisions anteriors d'altres veïns degut a que funciona per 'Routing by rumour' i no és capaç de tenir la informació de tota la topologia de la xarxa.

Altres protocols d'encaminament com OSPF i IS-IS són del tipus estat-enllaç i aleshores coneixen la topologia de la xarxa, calculant la millor ruta a través de Dijkstra. Si bé EIGRP no calcula la millor ruta, el fet de què no utilitzi Dijkstra ni conegui la topologia de la xarxa, permet tenir un consum de recursos menor a més a més de ser més escalable.

- **Falta d'una major parametrització:** EIGRP consta d'una mètrica composta on veritablement els únics valors que poden afectar la tria de rutes són el delay i el bandwidth. Al món del trànsit, un carrer es podria modelar per: el nombre de carrils (capacitat), velocitat mínima o màxima, ocupació actual, etc.

Si bé EIGRP compta amb un mecanisme de balanceig de trànsit (el qual resulta útil), només es propaga informació quan es produeixen canvis. Per tant no podem saber en tot moment l'ocupació d'un determinat enllaç per exemple. Tot i així, cap protocol està pensat amb una finalitat semblant.

CAPÍTOL 2. MILLORES DEL SISTEMA DE GUIATGE

Aquest capítol es basarà en l'estudi i implementació d'una sèrie de propostes de millora a través d'una aplicació programada en llenguatge Python. Aquesta aplicació té l'objectiu de millorar i estudiar algunes limitacions experimentades tant de l'entorn IPv4/IPv6, com del protocol d'encaminament dinàmic.

Principalment ens centrarem en dues limitacions:

- **Rutes no òptimes d'EIGRP:** proposem l'algoritme de cerca A* com una solució adequada en consum de recursos sense deixar enrere l'eficiència.
- **Falta d'una major parametrització:** com s'ha comentat al capítol anterior, EIGRP només permet una afectació de rutes mitjançant la modificació dels paràmetres delay i bandwidth. Aquests paràmetres no són suficients per modelar el trànsit, per tant proposem un estudi de rutes en funció de 3 indicadors: menor temps d'arribada possible, mínima distància i per últim evitació de camins saturats.

Aquests indicadors tindran aspectes presents com: velocitat dels enllaços, distàncies i temps. Per tal d'analitzar aquesta sèrie de factors, es dissenyarà un escenari amb una porció de carrers de Barcelona on l'usuari podrà introduir les coordenades origen i destí i aleshores obtenir el millor camí en funció de les circumstàncies.

2.1. Algoritme de cerca A*

Un dels inconvenients que presentava EIGRP era l'ús de rutes no sempre eficients a canvi d'un moderat cost computacional en comparació amb l'ús de l'algoritme Dijkstra en el cas d'OSPF i IS-IS. Si bé Dijkstra sí que troba les rutes òptimes (Shortest Path), no resulta viable en un escenari a gran escala degut al seu cost computacional de caire exponencial.

En aquest projecte proposem l'ús de l'algoritme de cerca A*:

A* [14] és una variant de Dijkstra [15], amb la diferència que A* tracta de buscar el millor camí mitjançant l'ús d'una funció *heurística* que dona prioritat als nodes que se suposa que són millors que d'altres, mentre que Dijkstra simplement explora tots els camins possibles (amb el consegüent alt cost computacional).

L'heurística és una estimació optimista del cost del camí més econòmic des d'un node donat fins al node objectiu.

A la Figura 2.1 representem una heurística molt senzilla, establint el millor camí com una línia recta entre A i C (tot i que fins i tot no existeixi camí directe a la realitat) i un possible camí real com A-B-C. La funció heurística és configurable i al cap i a la fi és una certa intuïció que recolza l'algoritme de cerca.

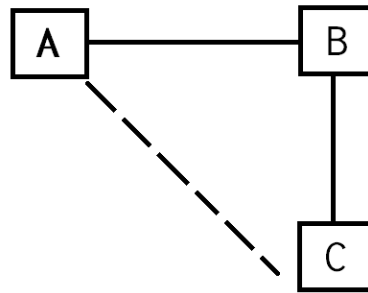


Figura 2.1: Exemple simple d'heurística

Per tant, la seva eficiència depèn de la funció heurística utilitzada, de manera que A* podria retornar un resultat no òptim a causa d'una mala heurística. De la mateixa manera, quant millor és aquesta, millor serà el resultat i menor serà el cost computacional.

Està dissenyat per ser més ràpid que Dijkstra, fins i tot si es requereix més memòria i més operacions per node, ja que explora molts menys nodes, i per tant és molt eficient.

A* utilitza la següent funció d'avaluació:

$$f(N) = g(N) + h'(N)$$

On:

- **f(N)** és el cost estimat del camí més curt fins a N.
- **g(N)** correspon al cost real del camí recorregut fins a arribar al node final N.
- **h'(N)** representa el valor heurístic admissible a evaluar des del node inicial, fins el node final N.

Si s'utilitza una funció heurística $h(N) = 0$ per tot N, s'obté que A* esdevé Dijkstra [16], de manera que buscaria totes les possibles solucions des d'un determinat node.

A* manté dues estructures de dades auxiliars que podem denominar *oberts*, implementat com una cua de prioritat (ordenada pel valor $f(N)$ de cada node), i *tancats*, on es guarda la informació dels nodes que ja han estat visitats.

A cada pas de l'algorisme, s'expandeix del primer node que estigui en oberts, i en cas que no sigui un node objectiu, calcula la $f(N)$ de tots els seus fills, els insereix en oberts, i passa el node avaluat a tancats.

En quant al **cost computacional**, aquest està íntimament relacionat amb la qualitat de l'heurística utilitzada:

En el pitjor dels casos, amb una heurística pèsima la complexitat esdevindrà exponencial (com Dijkstra, heurística inexistent), mentre que en el millor dels casos, amb una bona heurística esdevendria de manera quasi lineal.



Figura 2.2: Exemple de Shortest path A* vs Dijkstra

Font: <https://github.com/kevinwang1975/PathFinder>

A la Figura 2.2 s'observa un exemple de cerca de *Shortest Path* executat amb l'algorisme A* a l'esquerra, i Dijkstra a la dreta.

Els resultats obtinguts són: 113 nodes visitats (requadres blaus) amb A*, i de 614 nodes visitats amb Dijkstra. Es pot apreciar com A* esdevé molt més eficient, amb la diferència que A* no obté exactament el millor camí en aquest cas.

2.2. Explicació del procediment i les funcions a implementar

Per a l'elaboració d'aquestes seccions, s'ha sintetitzat informació de les fonts [17], [18], [19] i [20]. L'aplicació comportarà els següents procediments:

- Crear un escenari representatiu d'alguns carrers de Barcelona i definir velocitats màximes d'enllaços, distàncies i temps.
- Establir adjacències entre els nodes, és a dir, representar l'escenari de manera matricial.
- Si l'usuari indica coordenades, cercar el node més proper a partir d'unes coordenades inicials, així com el node final a partir de coordenades finals.
- Calcular els costos entre cada node tenint en compte criteris de menor temps, mínima distància i saturació.
- A partir d'aquests possibles nodes, costos, i informació de l'escenari, guiar l'algorisme a partir del càlcul d'heurística, definir el cost real dels enllaços i definir la funció d'avaluació $f(N) = g(N) + h'(N)$.
- Expandir els nodes, és a dir, contemplar totes les possibilitats en quant a nodes fills.
- Obtenir la llista dels nodes fills finals junt amb la seva corresponent informació, conformant el camí més òptim.

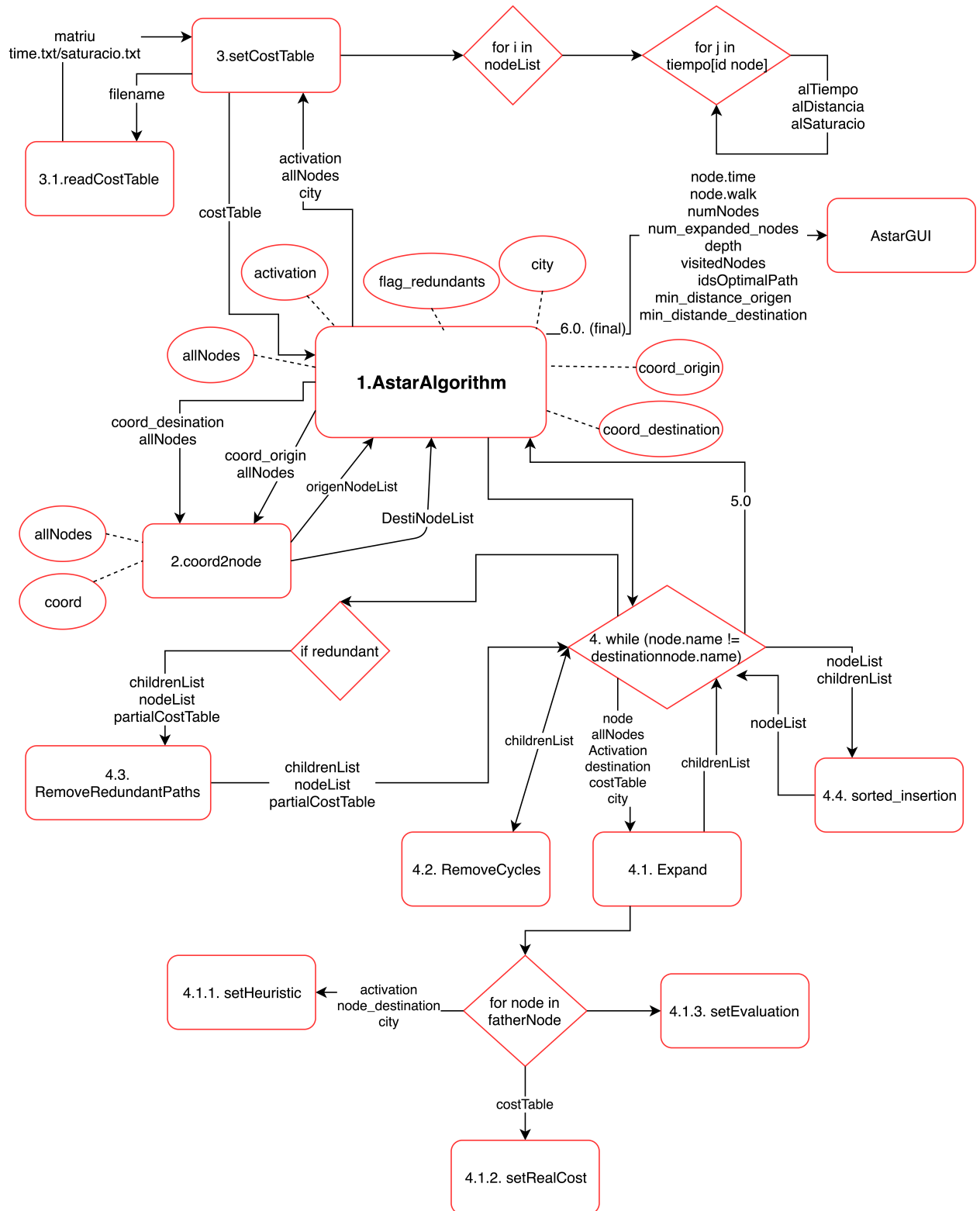


Figura 2.3: Diagrama de fluxe de l'aplicació en Python

2.2.1. Escenari a implementar

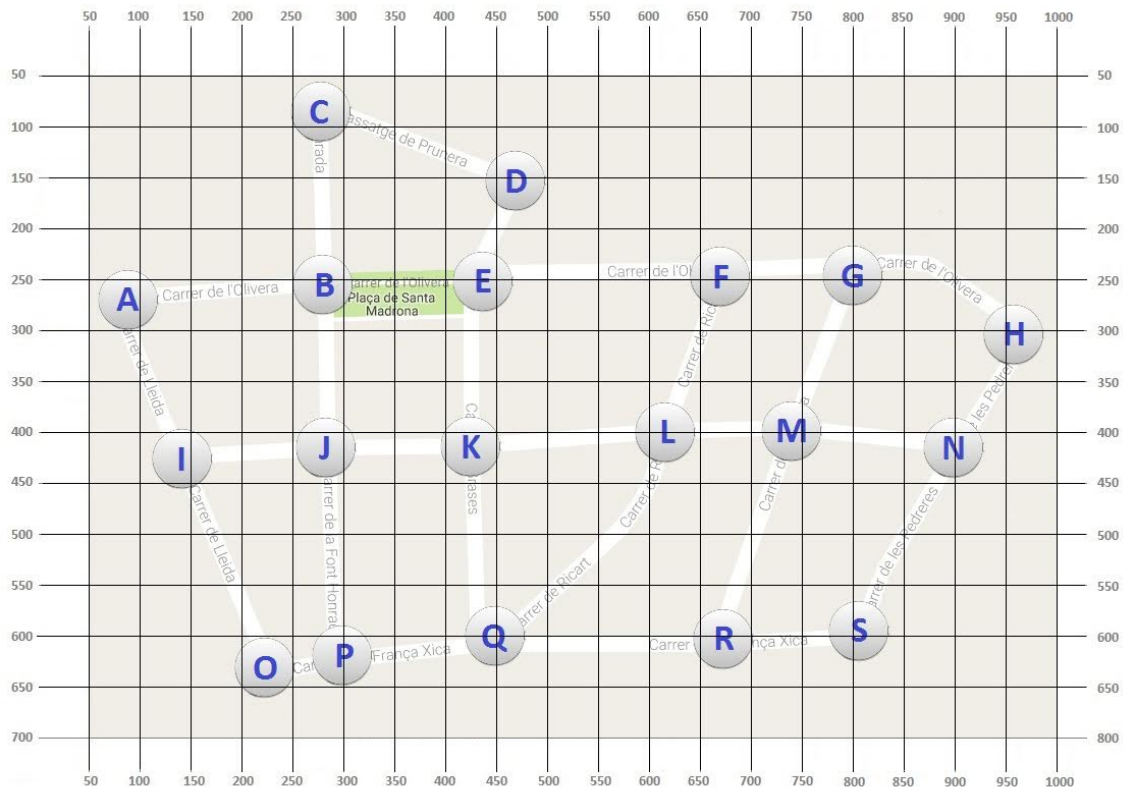


Figura 2.4: Carrers de Barcelona mapejats amb coordenades i nodes

- Enllaços **J-K** i **R-M** poden assolir una velocitat màxima de **20Km/h**.
- Enllaços **Q-L-F** poden assolir una velocitat màxima de **40Km/h**.
- Tota la resta d'enllaços assoleixen **50Km/h** com a màxim.

Aquesta decisió d'assignar diferents velocitats per cada enllaç ha estat per intentar imitar al màxim les condicions actuals. Entre els nodes J i K hi ha una església, i entre els nodes R i M un col·legi, pel que es pretén garantir seguretat en aquest tipus de zones més peatonals, donat que no tots els enllaços tenen les mateixes característiques.

2.2.2. Representació matricial de l'escenari

Els arxius així com les funcions esmentades a les següents seccions són a l'**apèndix E**.

2.2.2.1. Informació dels nodes

Cada node té el seu identificador (A, B, etc.), unes coordenades que l'ubiquen i per últim una velocitat associada que correspon al cap i a la fi a un tipus d'enllaç o 'línia':

Tipus 1 (50Km/h), tipus 2 (40Km/h) i tipus 3 (20Km/h). Aquesta informació es troba als arxius *Nodes.txt* i *InfoVelocity.txt*.

2.2.2.2. Matriu d'adjacència

Per tal de processar l'escenari, serà necessari representar les adjacències entre els nodes de manera matricial. Consisteix simplement en una matriu on les files i les columnes corresponen als nombres dels nodes, sent un '1' una adjacència directa amb un altre node. A l'arxiu *Connections.txt* hem fet constar aquesta representació matricial.

2.2.2.3. Matriu de temps

Per tal de representar les característiques físiques de cada enllaç tals com la distància entre cada node, dividirem la distància (metres) obtinguda a través de Google Maps entre la velocitat (convertida en m/s) i construirem una matriu molt semblant a la d'adjacència, però amb el temps que suposa recórrer cada enllaç (influint per tant a l'hora de calcular els costos). Correspon a l'arxiu *Time.txt*.

2.2.2.4. Matriu de saturació

Representarem la saturació (i per tant inviabilitat) d'un node amb una última matriu on simplement un '1' representa un node saturat, el qual suposarà un cost idealment infinit. Correspon a l'arxiu *Saturation.txt*.

2.2.3. Cerca del node més proper

Aquesta funció s'encarrega de trobar la distància entre dues coordenades cartesianes introduïdes per l'usuari i la distància real amb un node de la ciutat. és cridada sota el nom *coord2node*.

En aquesta ocasió, els paràmetres d'entrada són les coordenades introduïdes per l'usuari i una llista dels nodes que hi ha a la ciutat. En aquesta ocasió el càlcul també és simple: primerament es calcula la distància entre tots els nodes i les coordenades entrades, i es guarda en una llista amb l'ID del node i la distància.

Finalment s'ordenen els valors de les distàncies de menor a major i es guarden en una nova llista les identifikacions dels nodes de la llista ordenada.

2.2.4. Heurística

És cridada sota el nom *setHeuristic* i els paràmetres d'entrada són: el tipus de preferència, el node destinació i la informació de la ciutat on es troba el viatger.

Definim dues heurístiques principals:

- **Heurística de mínim temps:** aquesta heurística estudiarà el cas de viatjar en el menor temps possible fent ús de la coneguda fórmula temps=espai/velocitat però modificant la velocitat a la màxima de la ciutat (50Km/h) per fer una aproximació, obtenint una heurística admissible i optimista.

$$h(\text{temps}) = \frac{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}}{\text{Velocitat}_{\text{MAX}}}$$

- **Heurística de mínima distància:** en aquest cas serà l'arrel quadrada de les diferències de les distàncies al quadrat entre les dues coordenades cartesianes.

$$h(\text{dist.}) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Si bé resulta obvi que ambdues heurístiques estan relacionades, preferim separar-les donat que per exemple en un moment determinat podríem preferir un camí lent degut a la limitació de la velocitat de la via però curt per tal d'estalviar autonomia, o bé el camí més ràpid com acostuma a ser habitualment.

Aquestes heurístiques són el resultat de l'anomenada distància Euclidiana, que es dedueix a partir del teorema de Pitàgores, de manera que el resultat final és una línia directa entre el node origen i el node destí que faran la funció de guiar a l'algoritme per tal que expandeixi el mínim nombre de nodes.

Aquesta funció és una de les més importants de l'algorisme donat que definirà el comportament de l'algoritme en les aproximacions successives als diferents camins a seguir.

2.2.5. Cost

Es divideix realment en dues funcions: *setRealCost* i *setCostTable*.

En el primer cas, *setRealCost* s'encarregarà d'una de les dues grans parts de l'algorisme de cerca A*: el càlcul del **cost acumulatiu** entre dos nodes diferents.

El seu únic paràmetre d'entrada és un diccionari de dues dimensions que representa la matriu de costos, permetent així un càlcul del cost real més eficient. Actualitzarà el valor de la variable global 'g' comprovant el cost entre el node origen i el node final.

En canvi, la segona funció, *setCostTable* serveix per crear el diccionari que rebrà *setRealCost*. Finalment, es llegeix la taula de costos ja rebuda anteriorment mitjançant la funció *readCostTable*.

2.2.6. Heurística global

Aquesta funció és el resultat de les dues anteriors funcions i és cridada sota el nom *setEvaluation*. Simplement farà el següent càlcul: $f = g(\text{node}) + h$.

Com s'observa, només suma els valors del pes del node actual al del valor de l'heurística. No té cap paràmetre d'entrada i només realitza aquest càlcul.

2.2.7. Expansió d'un node

Aquesta funció buscarà els fills del node on es troba el programa actualment. És a dir, es buscaran els nodes relacionats a un en concret i que es trobin a distància 1 per crear-ne una llista d'adjacència. És cridada sota el nom *Expand*.

En aquesta ocasió, els paràmetres d'entrada són el node a expandir, una llista amb tots els nodes de la ciutat, el node destinació, el diccionari calculat a *setCostTable* i la informació de la ciutat.

El procediment és senzill: es crea una llista buida on s'introdueixen els ID dels elements del diccionari. Així s'obté una llista amb tots els nodes circumdants a un altre.

2.2.8. Eliminar cicles

S'encarrega de realitzar un altre pas important per l'algorisme A^* : l'eliminació de cicles que puguin provocar que l'algorisme entri en un bucle d'anàlisi del qual no pot sortir. Així doncs serà cridada sota el nom *RemoveCycles*, i el seu únic paràmetre d'entrada és una llista amb tots els nodes que fan referència a un altre, és a dir, la llista que crea la funció *Expand*.

El seu funcionament és senzill: primer es crea una llista de nodes fills d'un node. A continuació s'itera fins el final comprovant que no hi hagi cap fill que es trobi també en la llista de pares d'aquest node.

Es pot observar el seu corresponent diagrama de fluxe a la Figura 2.5.

2.2.9. Eliminar camins redundants

La penúltima funció bàsica també persegueix un objectiu semblant a la de l'anterior.

En aquest cas, però, es vol mirar d'eliminar els camins redundants (camins que porten a nodes ja visitats prèviament i que, lògicament, no són òptims) d'una llista *childrenList* que serà passada com a paràmetre.

Es pot observar el seu corresponent diagrama de fluxe a la Figura 2.6.

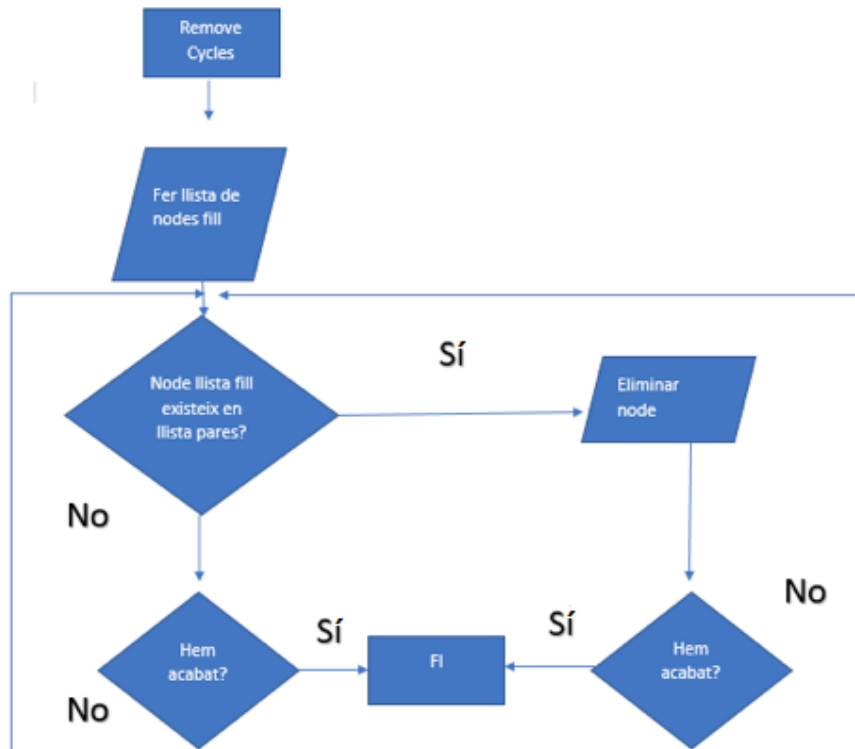


Figura 2.5: Diagrama de fluxe de la funció RemoveCycles

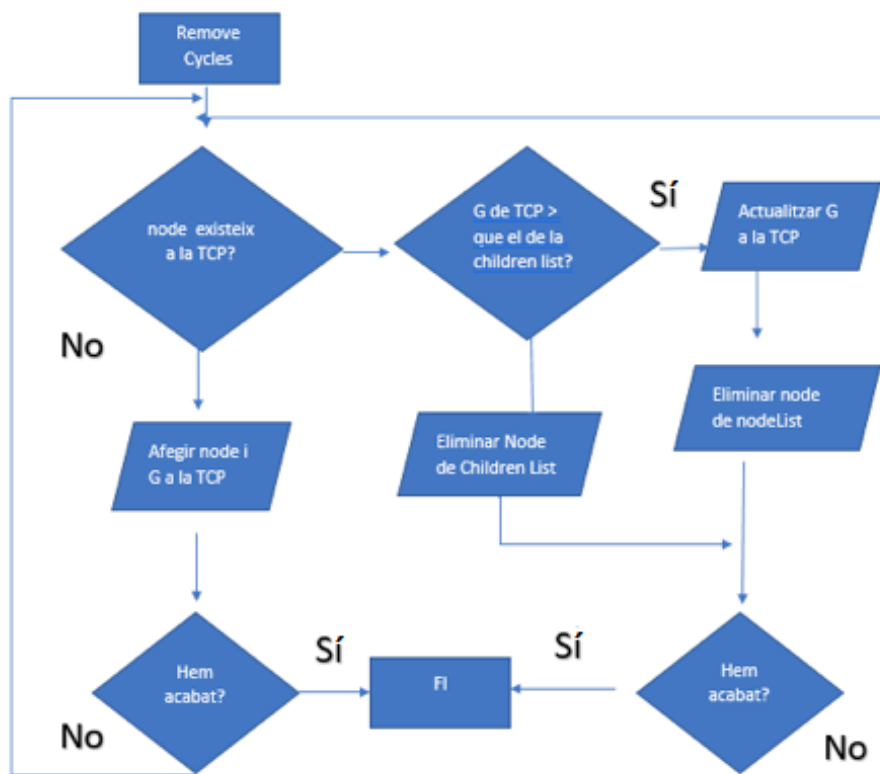


Figura 2.6: Diagrama de fluxe de la funció RemoveRedundantPaths

2.2.10. Inserció ordenada

L'última funció que s'ha implementat és la *sorted insertion*. Aquesta, tal i com el seu nom indica, insertarà ordenadament (seguint el valor f) cada node d'una llista que s'ha passat com a paràmetre (*childrenList*) en una nova llista que també s'ha passat com a paràmetre (*nodeList*).

Així doncs, mitjançant bucles *for* i *while*, s'aniran ordenant els nodes en funció del seu valor f , comprovant cas a cas que el valor de f sigui major que una certa posició de la llista a retornar i col·locant-lo tot just abans.

2.3. Anàlisi de rutes

Un cop explicat el funcionament del programa, ens centrarem en quines rutes proposa l'algoritme A* un cop establertes les següents condicions:

- Enllaços **J-K** i **R-M** a **20Km/h**, **Q-L-F** a **40Km/h** i resta de nodes a **50Km/h** com a màxim.
- Ús d'heurística de mínim temps i ús d'heurística de mínima distància, segons la preferència que indiqui l'usuari.
- Enllaç **A-I saturat**.

Amb aquestes condicions inicials, la funció *SetRealcost* obté el següent mapa de costos per a l'**heurística de temps**:

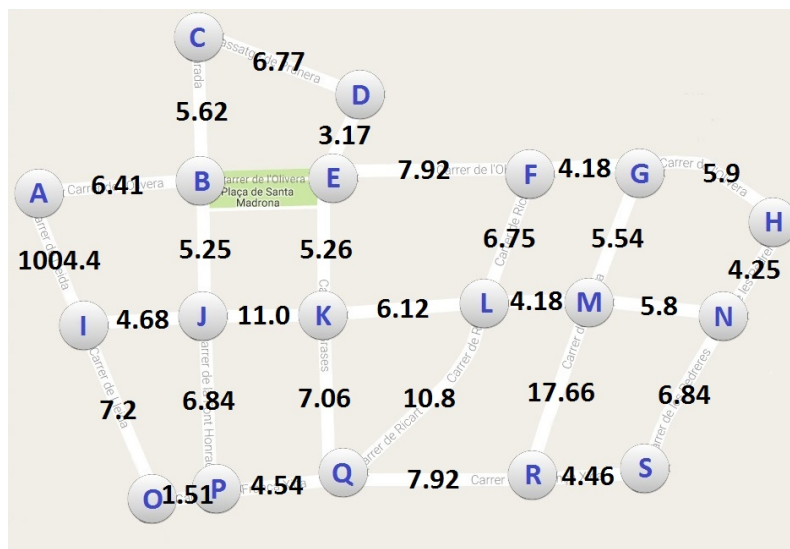


Figura 2.7: Costos de temps obtinguts per SetRealcost

De la mateixa manera, la funció *SetRealcost* obté el següent mapa de costos per a l'**heurística de distància**:

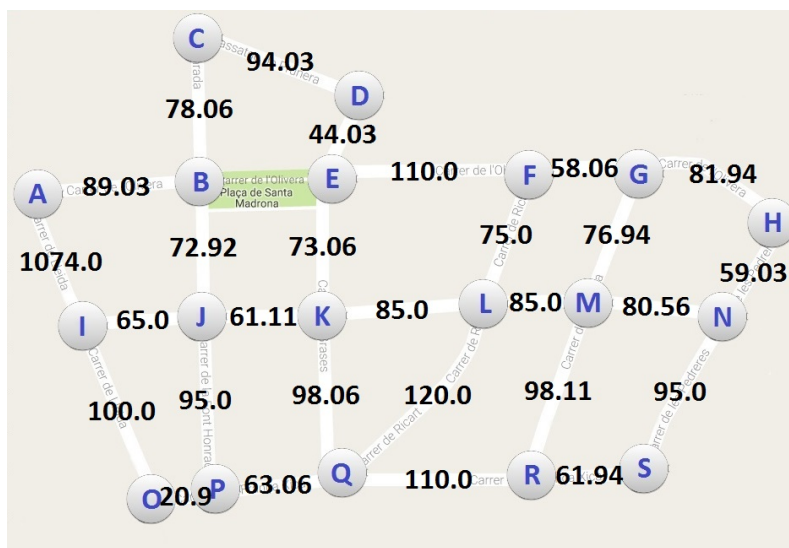


Figura 2.8: Costos de distància obtinguts per SetRealcost

A la Figura 2.9 es motra l'interfície gràfica que mostra el programa durant la seva execució:

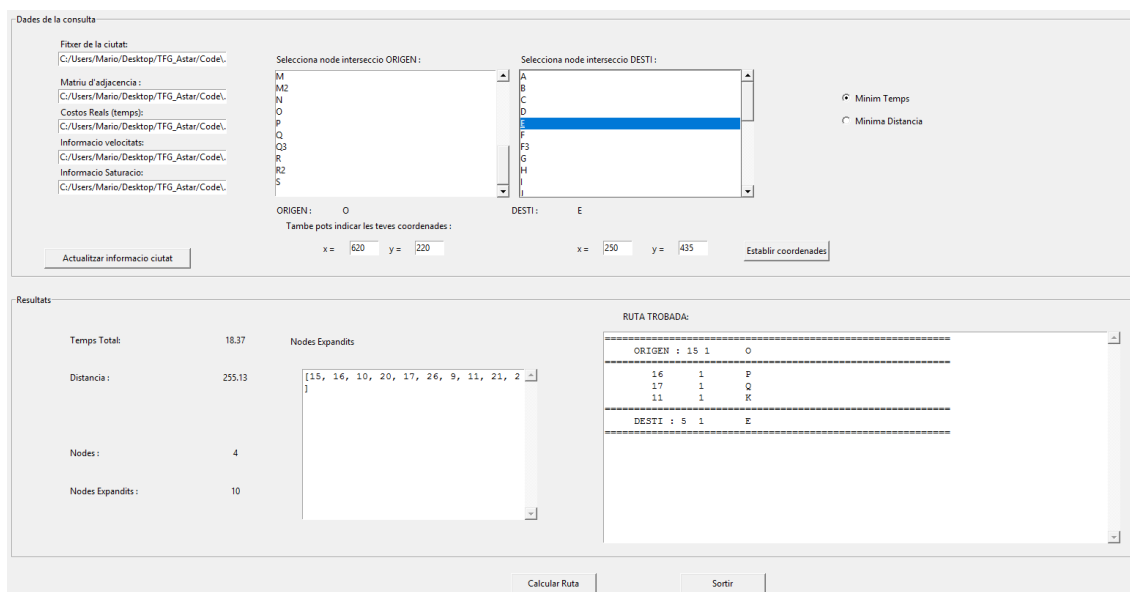


Figura 2.9: Interfície gràfica de l'aplicació

El funcionament és el següent:

L'usuari introdueix una sèrie de coordenades origen i destí, o bé indica directament el node origen i el node destí.

A continuació tenim dos possibles opcions, '**Mínim temps**' o '**Mínima distància**' en funció de quina preferència volem que s'apliqui al calcular la ruta. Finalment clicarem a '**Calcular ruta**'.

Un cop calculada la ruta, a la part inferior s'indica com a resultats:

El **temps total** del recorregut en segons, la **distància total** del recorregut en metres, el **nombre de nodes** que conformen la ruta, el nombre de **nodes expandits**, i per últim, el més important, la **ruta establerta** de principi a fi amb el corresponent nom de cada node.

2.3.1. Ruta A-I

Per començar a provar el correcte funcionament de l'algoritme, la primera ruta bàsica a analitzar és des del node A fins al node I, sent que es tracta precisament d'un camí saturat. L'algoritme A* ens retorna la següent ruta:

=====		
ORIGEN :	1	1 A
=====		
2	1	B
10	1	J
=====		
DESTI :	9	1 I
=====		

La simulació ha retornat les següents dades:

- **Temps total** del recorregut: 16.34s
- **Distància total** del recorregut: 226.94m
- Nombre de **nodes recorreguts**: 3
- Nombre de **nodes expandits**: 4 (heurística temps) i 3 (heurística distància)
- **Temps d'execució**: 0.001684 segons

Com podem observar, ha realitzat la ruta més òptima, donat que el recorregut A-I directe hagués suposat un cost de 1004.4 amb preferència de temps o bé un cost de 1074 amb preferència de distància, sent que en canvi el recorregut A-B-J-I suposa un cost total de 20.01 en el primer cas, i de 277.85 en el segon cas.

Per tant, independentment de l'opció triada, mai voldrà passar per l'enllaç saturat, que és precisament l'objectiu d'haver atorgat un cost tan elevat a l'enllaç A-I, sent l'única diferència el nombre de nodes expandits.

De fet, dels 26 nodes establerts només n'ha expandit 4 en heurística de temps i 3 amb la de distància, amb un temps de còmput d'aproximadament 1.68 mil·lisegons.

2.3.2. Ruta G-R

2.3.2.1. Heurística de menor temps

Una altra ruta interessant per analitzar és el recorregut des de G fins a R amb aquesta determinada heurística.



Figura 2.10: Ruta G-R amb preferència de temps

La simulació ha retornat les següents dades:

- **Temps total** del recorregut: 21.45s
- **Distància total** del recorregut: 297.91m
- Nombre de **nodes recorreguts**: 4
- Nombre de **nodes expandits**: 11
- **Temps d'execució**: 0.00169 segons

Com podem observar, l'algoritme ha preferit evitar l'enllaç M-R donat que estava limitat a tan sols 20Km/h, i per tant ha preferit 'vorejar' per tal de realitzar el recorregut en un temps menor. Ha actuat correctament donat que el cost d'haver fet la ruta G-M-R suposava un cost de 23.2 a diferència dels 21.45 amb aquesta ruta.

2.3.2.2. Heurística de menor distància

En aquest cas l'heurística dóna una certa preferència a un recorregut que suposi una menor distància tot i que aquest fet impliqui un major temps de recorregut. Obtenim la següent ruta:

2.3.3. Ruta O-F

2.3.3.1. Heurística de menor temps

Amb aquesta determinada heurística, obtenim la següent ruta:

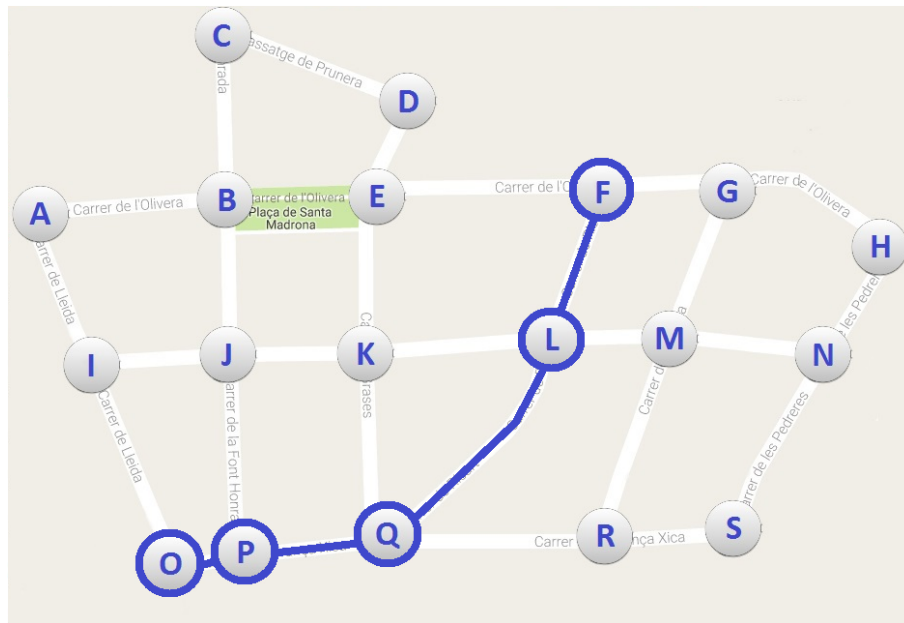


Figura 2.12: Ruta O-F amb preferència de temps

La simulació ha retornat les següents dades:

- **Temps total** del recorregut: 23.60s
- **Distància total** del recorregut: 279.02m
- Nombre de **nodes recorreguts**: 4
- Nombre de **nodes expandits**: 16
- **Temps d'execució**: 0.00182 segons

La ruta resultant és correcta; prefereix passar pels enllaços Q-L-F tot i estar limitats a 40Km/h, donat que tot i així segueix esdevenint el camí més ràpid.

Per últim, com podem observar, en aquest cas ha expandit 16 nodes per tal de trobar la solució; això és degut a la major complexitat que li pot presentar a l'algoritme trobar el camí més òptim donat que aquest recorregut era més difícil que els anteriors.

En aquest cas ha hagut de considerar un major nombre de nodes per tal de trobar la solució, tot i que seguirà sent menor al nombre de nodes que expandeixi Dijkstra.

2.3.3.2. Heurística de menor distància

Amb aquesta determinada heurística, obtenim la següent ruta i dades resultants:



Figura 2.13: Ruta O-F amb preferència de distància

- **Temps total** del recorregut: 25.98s
- **Distància total** del recorregut: 342.08m
- Nombre de **nodes recorreguts**: 5
- Nombre de **nodes expandits**: 7
- **Temps d'execució**: 0.00179 segons

S'ha proposat l'anàlisi d'aquesta ruta amb un objectiu en concret: anteriorment s'ha explicat que l'algoritme A* és més òptim que Dijkstra donat que expandeix molts menys nodes i computacionalment és més eficient, però precisament pel motiu que no expandeix tots els nodes, no sempre troba la millor ruta.

Aquest cas és precisament un bon exemple:

Donat que l'heurística estableix una línia 'directa' entre O i F, l'algoritme s'ha deixat guiar per aquesta consideració i per tant, un cop al node Q, ha considerat que entre Q i F, l'enllaç Q-L tenia més distància en comparació amb Q-K, que a més a més concordava més segons l'heurística configurada. Per tant A* ha considerat la ruta O-P-Q-K-L-F amb el node 'K' intermig.

Si bé no és la ruta més òptima, computacionalment ha resultat molt lleuger amb només 7 nodes expandits i 1.79 mil·lisegons d'execució.

2.3.4. Comparació amb Dijkstra

Finalitzarem aquest capítol amb una comparació ràpida sobre com actua Dijkstra en comparació amb A*. Per tal d'executar Dijkstra en aquesta aplicació és tan senzill com establir una heurística = 0, de manera que l'algoritme no té cap tipus d'ajuda i haurà d'expandir una gran part dels nodes per tal de trobar el *best path*.

En el cas d'aquesta última ruta O-F, amb Dijkstra obtenim els següents resultats:

- **Temps total** del recorregut: 23.60s
- **Distància total** del recorregut: 279.02m
- Nombre de **nodes recorreguts**: 4
- Nombre de **nodes expandits**: 21
- **Temps d'execució**: 0.00202 segons

=====			
ORIGEN :	15	1	0
=====			
16	1	P	
17	1	Q	
26	3	Q3	
25	3	L3	
24	3	F3	
=====			
DESTI :	6	1	F
=====			

Nota: els nodes Q3, L3 i F3 són nodes 'ficticis' ubicats exactament en les mateixes coordenades que Q, L i F respectivament, i es van crear per tal d'indicar que aquests enllaços són del tipus 3 i per tant tenen una velocitat màxima equivalent a 40Km/h entre ells.

Com hem pogut observar, Dijkstra ha realitzat la ruta O-P-Q-L-F que correspon a la mateixa que A* va conformar en el cas de seleccionar l'heurística que tenia com a preferència un menor temps.

La diferència rau principalment en què **ha expandit 21 nodes** en lloc dels 16 d'A* amb heurística de menor temps i 7 amb la de menor distància, a més a més d'un lleuger major temps d'execució en comparació amb A*.

Si bé en aquest petit escenari les diferències no són tan notables, en el cas de tenir un escenari inmens com podria ser la ciutat de Barcelona sencera, el cost computacional de Dijkstra acabaria esdevenint exponencial.

També hem de tenir en compte que sense l'ajuda d'una heurística que el guiï, el nombre de nodes recorreguts podria ser molt elevat, a més a més d'un temps de còmput considerable com anteriorment vam poder observar a la Figura 2.2. En el cas d'implementar A* a gran escala, una de les claus rauiria en configurar una bona heurística per tal de reduir la complexitat computacional.

CAPÍTOL 3. CONCLUSIONS I LÍNIES DE FUTUR

3.1. Conclusions generals

Al llarg d'aquest projecte s'ha desenvolupat el concepte d'un nou model de sistema de guiatge alternatiu i s'han presentat els seus avantatges, dificultats i lògicament, limitacions.

El procés d'estudiar la implementació d'un sistema de guiatge per a vehicles autònoms en entorn xarxa ha comportat l'aparició d'idees molt presents actualment, com ara la monitorització de tot el que ens envolta, és a dir, la necessitat de tenir tot connectat a la xarxa per tal de gaudir de flexibilitat, fàcil accés i centralització.

Com hem pogut observar al llarg del document, aspectes com l'urbanisme actual, els senyals de trànsit, o fins i tot la 'direcció prohibida' han resultat producte d'un consens i adaptació de l'automòbil clàssic a unes infraestructures que en el moment de sorgir el gran invent de l'automòbil, van resultar creacions lògiques i adients. Aquest projecte ha anat més enllà, proposant una remodelació del model actual, i per tant una anàlisi del concepte de xarxa de trànsit.

La implementació d'un sistema de guiatge sota el protocol d'encaminament dinàmic EIGRP [3] ha comportat alguns avantatges tals com: ràpida convergència al efectuar-se canvis de topologia, possibilitat de balancejar la càrrega entre els enllaços, monitorització de l'estat d'un enllaç, i sobretot una alta escalabilitat; sent possible treballar amb grans escenaris sense requerir excessives especificacions de hardware.

Si bé l'entorn xarxa ha resultat prou extrapolable, també ens hem trobat amb certes limitacions. Principalment l'exhauriment d'adreces IPv4 [12] (amb IPv6 com a possible solució), les rutes no òptimes d'EIGRP, i finalment una falta d'una major parametrització en quant a mètrica i conformació de rutes. Són aquests els motius principals pels qual vam decidir elaborar una aplicació de navegació de trànsit en llenguatge Python que ens permetés analitzar aquests aspectes.

Aquesta aplicació tenia com a objectiu analitzar l'A* [14] com a protocol de cerca de Best path, així com representar algunes condicions actuals tals com velocitats diferents segons el tipus de via i el seu voltant, així com enllaços saturats que s'han d'evitar. El resultat ha estat prou satisfactori, obtenint millors resultats que Dijkstra gràcies a l'ús de l'heurística, assolint aleshores un equilibri entre rutes òptimes i cost computacional.

La clau de l'efectivitat d'aquest algoritme ha estat la configuració d'unes certes heurístiques que ajudaven a l'algoritme de cerca a tenir una certa intuïció o direcció de la solució final. L'ús de l'heurística no es tracta d'una novetat, sino que s'ha utilitzat des dels inicis de la informàtica, posant com a exemple el cas d'Alan Turing desxifrant la màquina Enigma.

A més a més, ha estat possible observar com actua l'algoritme en front a preferències de l'usuari com per exemple trobar el recorregut més ràpid, o bé el que suposés una mínima distància segons les heurístiques aplicades. Aquest plantejament va resultar interessant donat que si bé habitualment acostumem a preferir la ruta més ràpida, en condicions de

baixa autonomia potser és preferible una ruta amb una menor distància (però sempre abolint els enllaços saturats).

Les conclusions principals que s'han pogut extreure d'aquest projecte són les següents:

- El sistema de guiatge plantejat en aquest projecte no suposa que un futur plantejament impliqui basar-lo en xarxa totalment, ni potser començar des de 0 un software completament especialitzat, sino que l'objectiu ha estat plantejar fins a quin punt és extrapolable el trànsit vehicular a l'entorn xarxa i per tant, argumentar que és possible aprofitar molts avantatges que implica una topologia d'aquest tipus, però sense perdre de vista que compta amb limitacions.
- En quant a la seva viabilitat, tot i haver treballat amb escenaris petits, amb conceptes tractats al llarg del projecte com ara la sumariació de rutes o bé l'ús d'A* com a protocol de cerca del millor camí, es pretén que els requeriments a nivell computacional no siguin alts. Durant les proves de l'escenari implementat en Python, s'ha pogut observar com sovint A* trobava la solució òptima expandint menys nodes que Dijkstra en funció del bon plantejament de l'heurística o no. I en cas de no trobar exactament la solució òptima, s'ha apropiat molt, justificant una menor precisió però més eficient.
- En relació al punt anterior, tota balisa instal·lada a cada intersecció de Barcelona s'espera que tingui unes especificacions molt moderades. Totes les simulacions han estat rondant els 1.7 mil·lisegons aproximadament en quant a temps d'execució del programa. Si bé és cert que no és comparable el hardware d'un ordinador de característiques més avançades als d'un possible router més limitat, no s'espera que els temps de càlcul siguin elevats ja que un dels motius d'haver implementat A* és degut a la seva eficiència en grans escenaris.
- La tecnologia relacionada amb el món del vehicle s'està desaprofitant ja que es basen en la infraestructura actual. Com s'ha comentat en el document, els vehicles autònoms estan fets per conviure amb vehicles clàssics, respectant els senyals i actuant en conseqüència. Aquest plantejament si bé és l'adequat en temps presents donat que actualment és inviable realitzar un canvi tan gran (eliminar vehicles clàssics, reduir o eliminar senyalització), seguir les infraestructures actuals suposarà un límit de les possibilitats que aquest gran invent ens podria habilitar.
- Les conseqüències dels vehicles de combustió resulten letals per al medi ambient, es fa evident la necessitat de prescindir d'aquestes fonts d'energia perjudicials i optar per energies alternatives, així com reduir la taxa de mortalitat tant per pol·lució com per accidents de trànsit. Per tant, precisament la idea del nou model de trànsit podria ajudar a reduir quantitativament aquests grans problemes.

3.1.1. Impacte social

Si bé la possible adopció d'aquest sistema en la nostra societat comportaria una reducció de manera exhaustiva de les emissions de gasos nocius, i per tant obtenint un menor impacte medioambiental, sí que implicaria un cert impacte social necessari d'analitzar:

Actualment hi ha unes certes indústries fomentades al voltant de l'automòbil, tals com: el petroli, les empreses dedicades a la fabricació de components i vehicles en sí, mecànics, etc. I tampoc ens oblidem dels impostos en relació amb el vehicle: impost per matriculació, revisió tècnica del vehicle, multes de trànsit, etc.

Si bé la dependència del petroli s'està veient reduïda i està demostrat el seu negatiu impacte, altres aspectes com els relacionats a la direcció general de trànsit es fan més complicats. La idea proposada amb aquest canvi revolucionari en el món del transport, és el d'evitar que tot ciutadà tingui el seu vehicle personal a l'àrea urbana. És a dir, de la mateixa manera que existeix el transport públic i sovint és ineficient per el principal motiu de que no totes les rutes són equiprobables (p. ex autobusos en baix ús en determinades hores o condicions), tenir un vehicle personal també pot resultar altament ineficient.

Amb la invenció d'un nou model de sistema de guiatge dels vehicles autònoms, tot ciutadà podria pagar una tasa mensual del mateix tipus que el transport públic (de fet, això és un nou model de transport públic) i utilitzar-lo com un servei més. L'usuari introduiria les dades sobre la seva ubicació i el seu destí i aleshores compartiria vehicle amb persones interessades en un trajecte similar. Els vehicles podrien viatjar a mode de tren de carretera o *platooning* [35], i a més a més, cap usuari hauria de conduir ni prendre decisions. Les úniques carències serien presentades en àrees on no s'hagués implementat el sistema de guiatge, sent necessari per tant el lloguer de vehicles fora de nuclis urbans.

Lògicament, el fet d'abolir el vehicle privat implicaria una redistribució de les taxes actuals, com per exemple un impost per circulació més elevat. Tot i així, conseguiríem reduir el nombre de cotxes que circulen per la ciutat, ja que el nombre de persones transportades per vehicle seria major i per tant, implica també un consum de l'energia més responsable.

Per últim, amb el consegüent canvi de tecnologia de l'automòbil i la seva major semblança amb l'electrònica que la mecànica, molts oficis es veurien afectats, però en cap cas substituïts o eliminats.

La fórmula següent representa la identitat Sistema-Entorn [22]:

$$C_s + A_s = I_e + S_e$$

- C_s = Complexitat del sistema.
- A_s = Capacitat d'adaptació (anticipació) del sistema a l'entorn.
- I_e = Incertidumbre de l'entorn.
- S_e = Sensibilitat de l'entorn (impacte) del sistema.

És a dir, donada la complexitat d'un sistema, si la incertesa de l'entorn augmenta, es pot controlar la situació augmentant la capacitat d'anticipació del sistema (millor percepció, millor coneixement, etc.) o bé augmentant la capacitat d'influir sobre l'entorn, és a dir, amb més mobilitat o tecnologia [21].

Per tant, davant d'aquests canvis i incerteses de l'entorn, estarem parlant d'una transformació i adaptació de l'individu sota les noves condicions actuals per tal d'evitar excloure's del sistema actual. D'aquesta manera podrem avançar sense frenar evolucions de la societat.

3.1.2. Línies futures

Per concluir, aquesta petita secció pretén donar alguna idea futura sobre com s'espera un urbanisme més avançat. Un dels objectius d'aquest projecte ha estat el de minimitzar l'impacte medioambiental actual, tenint en compte aspectes com la qualitat de l'aire principalment així com el nivell de sonoritat.

A aquest aspecte podríem sumar el d'augmentar la quantitat de zones verdes i fer una ciutat per els vianants. Aquest objectiu es podria aconseguir de la següent manera: enlairant el trànsit.

La idea es recolzaria en el concepte *Automated Transit Networks* (ATN) [23], és a dir, seguim apostant per el transport automatitzat (per exemple el metro de la L9) però donant-se en estructures superiors, i establint-se com un punt mitg entre el metro i el vehicle, conseguint una ciutat completament diferent. És el concepte que descriu l'esborrany de la Figura 3.1.



Figura 3.1: Esborrany de ciutat amb el trànsit enlairat [23]

Font: ATN: A Review of the State of the Industry and Prospects for the Future, MTI Report 12-31

A més a més, el plantejament del trànsit enlairat podria comportar més avantatges com ara la incorporació de plaques solars a la seva infraestructura, fet que podria ajudar a reduir el consum.

Si bé aquest tipus d'implementació dista molt del concepte definit al llarg d'aquest projecte donat que no atorga la mateixa llibertat ni comoditat a l'usuari, és un factor a tenir en compte donat que cada cop es tendeix a acomodar la ciutat per als vianants, i aquesta podria ser una alternativa.

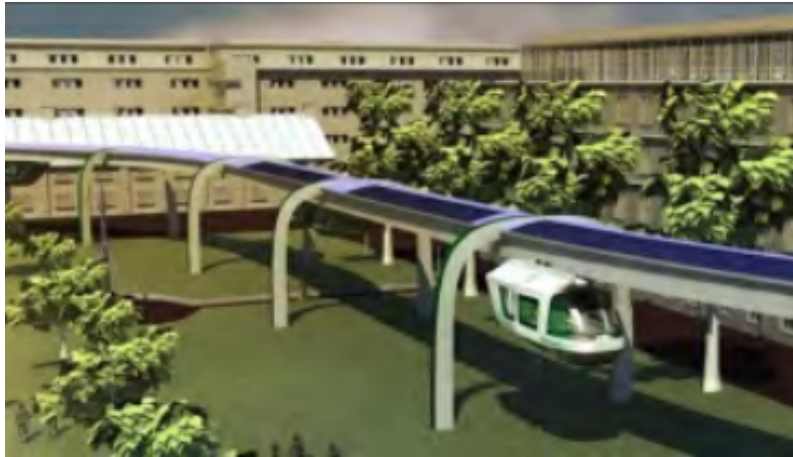


Figura 3.2: Maqueta visual d'ATN amb plaques solars [23]

Font: ATN: A Review of the State of the Industry and Prospects for the Future, MTI Report 12-31

BIBLIOGRAFIA

- [1] Rajkumar Buyya, Chee Shin Yeo, Srikumar Venugopal, James Broberg, and Ivona Brandic. Cloud computing and emerging {IT} platforms: Vision, hype, and reality for delivering computing as the 5th utility. *Future Generation Computer Systems*, 25(6):599 – 616, 2009.
- [2] E. Bouillet, D. Mitra, and K. G. Ramakrishnan. The structure and management of service level agreements in networks. *Selected Areas in Communications, IEEE Journal on*, 20(4):691–699, May 2002.
- [3] A. Predieri, M. Biancani, S. Spadaro, G. Bernini, P. Cruschelli, N. Ciulli, R. Monno, S. Peng, Y. Yan, N. Amaya, G. Zervas, N. Calabretta, H. Dorren, S. Iordache, J.C. Sancho, Y. Becerra, M. Farreras, C. Liou, and I. Hussain. Deliverable 2.2 - architecture design. Technical report, The Lightness project, 2013.
- [4] Mohammad Al-Fares, Alexander Loukissas, and Amin Vahdat. A scalable, commodity data center network architecture. In *Proceedings of the ACM SIGCOMM 2008 Conference on Data Communication*, SIGCOMM '08, pages 63–74, New York, NY, USA, 2008. ACM.
- [5] Infiniband architecture annex a 16: Roce. *InfiniBand Trade Association*, April 2010.
- [6] Infiniband architecture volume 1, release 1.2.1. *InfiniBand Trade Association*, January 2008.
- [7] IEEE standard for local and metropolitan area networks: Media access control (MAC) bridges. *IEEE Std 802.1D-2004 (Revision of IEEE Std 802.1D-1998)*, June 2004.
- [8] IEEE standard for local and metropolitan area networks—media access control (MAC) bridges and virtual bridges. *IEEE Std 802.1Q, 2012 Edition, (Incorporating IEEE Std 802.1Q-2011, IEEE Std 802.1Qbe-2011, IEEE Std 802.1Qbc-2011, IEEE Std 802.1Qbb-2011, IEEE Std 802.1Qaz-2011, IEEE Std 802.1Qbf-2011, IEEE Std 802.1Qbg-2012, IEEE Std 802.1aq-2012, IEEE Std 802.1Q-2012*, December 2012.
- [9] A. Dhamdhere, C. Dovrolis, and P. Francois. A value-based framework for internet peering agreements. In *Teletraffic Congress (ITC), 2010 22nd International*, pages 1–8, Sept 2010.
- [10] J. Chabarek, J. Sommers, P. Barford, C. Estan, D. Tsang, and S. Wright. Power awareness in network design and routing. In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, April 2008.
- [11] A. Tzanakaki, M. Anastasopoulos, K. Georgakilas, J. Buysse, M. De Leenheer, C. Develder, Shuping Peng, R. Nejabati, E. Escalona, D. Simeonidou, N. Ciulli, G. Landi, M. Brogle, A. Manfredi, E. Lopez, J.F. Riera, J.A. Garcia-Espin, P. Donadio, G. Parladori, and J. Jimenez. Energy efficiency in integrated IT and optical network infrastructures: The geysers approach. In *Computer Communications Workshops (INFOCOM WKSHPS), 2011 IEEE Conference on*, pages 343–348, April 2011.

- [12] R. Ramaswami, K. Sivarajan, and G. Sasaki. *Optical Networks: A Practical Perspective*. Morgan Kaufmann series in networking. Elsevier Science, 2009.
- [13] M. Allman, V. Paxson, and E. Blanton. TCP Congestion Control. RFC 5681 (Draft Standard), September 2009.
- [14] Fibre channel backbone 5. *ANSI/INCITS 462-2010*, May 2010.
- [15] D. Cohen, T. Talpey, A. Kanevsky, U. Cummings, M. Krause, R. Recio, D. Crupnicoff, L. Dickman, and P. Grun. Remote direct memory access over the converged enhanced ethernet fabric: Evaluating the options. In *High Performance Interconnects, 2009. HOTI 2009. 17th IEEE Symposium on*, pages 123–130, Aug 2009.
- [16] Spectral grids for wdm applications: Dwdm frequency grid. *ITU-T Recommendation G.694.1 (02/2012)*, February 2012.
- [17] Christoforos Kachris and Ioannis Tomkos. A survey on optical interconnects for data centers. *Communications Surveys Tutorials, IEEE*, 14(4):1021–1036, Fourth 2012.
- [18] W. Miao, N. Calabretta, H. Dorren, Y. Yan, G. Zervas, Y. Shu, E. Hugues-Salas, S. Yan, S. Peng, D. Simeonidou, S. Spadaro, and F. Agraz. Deliverable 3.1 - release of the design and early evaluation results of ops switch, ocs switch, and tor switch. Technical report, The Lightness project, 2013.
- [19] R. Jain and S. Paul. Network virtualization and software defined networking for cloud computing: a survey. *Communications Magazine, IEEE*, 51(11):24–31, November 2013.
- [20] Software-defined networking: The new norm for networks. *Open Networking Foundation*, April 2012.
- [21] Openflow switch specification 1.4.0. *Open Networking Foundation*, October 2013.
- [22] G. Bernini, P. Cruschelli, N. Ciulli, R. Monno, S. Spadaro, D. Careglio, S. Peng, A. Predieri, M. Biancani, C. Liou, and I. Hussain. Deliverable 2.2 - architecture design. Technical report, The Lightness project, 2012.
- [23] B. Jaumard, C. Meyer, B. Thiongane, and Xiao Yu. Ilp formulations and optimal solutions for the rwa problem. In *Global Telecommunications Conference, 2004. GLOBE-COM '04. IEEE*, volume 3, pages 1918–1924 Vol.3, Nov 2004.

APÈNDIX A. HISTÒRIA I EVOLUCIÓ DEL TRÀNSIT VEHICULAR

Aquesta secció a mode d'annex del projecte consisteix en un recull i anàlisi d'informació que tractaran els inicis de l'automòbil, passant pels canvis i concepcions noves a nivell d'urbanisme en conseqüència, i sobretot els efectes negatius tant a nivell medioambiental i mortal que ha ocasionat l'adopció d'aquest popular medi de transport.

Per últim, s'introduiran les ineficiències que l'actual model de trànsit suposa a nivell d'eficàcia i logística. L'objectiu és donar la raó de ser a aquest projecte, justificant la necessitat de l'aparició d'avenços tecnològics com el sistema de guiatge que s'ha plantejat al llarg del projecte.

A.1. Aparició de l'automòbil i conseqüències vials

El segle XX comença amb l'exitosa aparició d'un enginy que canviaria per sempre el món del transport i les comunicacions: l'automòbil. Aquest fet revolucionari també va portar el caos als camins, motiu pel qual es feia indispensable establir normes per a organitzar-lo i fer-lo segur.

A.1.1. Història de l'automòbil

La informació exposada en aquesta subsecció ha estat extreta de la font [24]:

Tot va començar amb la invenció de la màquina de vapor i el motor a gas a finals del segle XVII. Nicolas-Joseph Cugnot va desenvolupar el primer automòvil de vapor de Cugnot, pesat, de difícil conducció i orientat al trasllat d'artilleria.

El primer vehicle motoritzat a ser produït amb fins comercials va ser un automòbil de tres rodes. Aquesta innovació va ser produïda en 1885 per l'enginyer alemany Karl Benz i posseeix un motor a gasolina. Va ser anomenat *Motorwagen* (vehicle motoritzat), les primeres unitats van ser produïdes per l'empresa de l'inventor, la Benz Co, a la ciutat alemanya de Mannheim. Com a curiositat, aquest vehicle comptava amb el sistema d'arrencada a maneta i tenia potència de 0.8 CV, podent aconseguir 18 quilòmetres per hora.

Així doncs, neix el motor com una possibilitat comercial i donant lloc a la idea del automòbil actual.

Com a principals conseqüències a destacar:

- Totes les grans fàbriques que coneixem actualment, neixen en els últims quinze anys del segle XIX: Mercedes, Peugeot, Renault, Ford, Fiat, etc.
- Comença la revolució del transport en automòbil.



Figura A.1: El primer automòbil de Karl Benz [24]

- Els tallers es converteixen en fàbriques, implicant un canvi a nivell social en quant a model de treball.
- Apareix la necessitat d'inventar un codi de senyals de circulació.
- Degut a les necessitats físiques de circulació de l'automòbil es fa vital l'asfaltament de camins, implicant una important inversió a més a més d'un nou model d'urbanisme.
- Neixen les carreteres d'alta velocitat actualment conegudes com autopistes i autovies.

El principal gran problema dels seus inicis (com tot nou invent) va ser l'elevat preu que suposava la construcció d'un automòbil, fet que va millorar amb la introducció de les fàbriques en cadena.

Aquest fet va implicar una reducció progressiva en els preus i per tant en un augment de compradors junt amb un nou disseny d'urbanisme. Tret d'altres implicacions a nivell social i econòmic, on realment es centra aquesta secció és en l'inici d'un sistema de circulació que permetia en tot moment tenir el trànsit regulat reduint el perill dels conductors i els vianants.

A.1.2. Naixement de la senyalització transitària

La informació exposada en aquesta subsecció i la següent, ha estat extreta de la font [25]:

Els patrons bàsics dels senyals de trànsit es van establir al 1908 el Congrés Internacional de Carreteres, a Roma. En 1909, nou governs europeus van acordar l'ús de quatre símbols pictòrics, el que indica 'stop', 'corba', 'intersecció', i 'a nivell d'encreuament de ferrocarril'. L'intens ús i el creixement que els senyals de trànsit suporten entre 1926 i

1949 en paral·lel al creixement del parc automobilístic va conduir al desenvolupament del sistema de senyal de trànsit d'Europa.

El 1968, els països europeus van signar a la Convenció de Viena un tractat sobre circulació per carretera amb l'objectiu d'estandarditzar les normes de trànsit en els països participants, facilitar el trànsit internacional per carretera i augmentar la seguretat viària. Part del tractat sobre senyalització viària defineix els senyals de trànsit.

Com a resultat d'això, a Europa Occidental els senyals de trànsit estan ben estandarditzats, tot i que encara hi ha algunes excepcions específiques de cada país, que daten majoritàriament de l'època anterior a 1968. El principi de la norma europea per als senyals de trànsit és quines formes i colors es van a utilitzar per indicar fins idèntics. En l'actualitat el tractat compta amb 52 països signataris, en els quals s'inclou tot Europa, gran part d'Àsia i molts països africans.

Els Estats Units va desenvolupar el seu propi sistema de senyalització viària, que també va ser adoptat per altres nacions (amb variacions pròpies de cada país). Està implementat en tota Amèrica i gairebé la totalitat d'Oceania i alguns països africans.

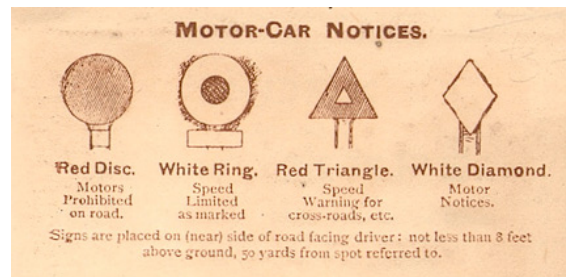


Figura A.2: Primeres senyals de trànsit dels Estats Units [25]

Actualment existeixen entre 150 i 200 senyals diferents depenent de l'interès que tingui cada país en tenir un trànsit ordenat, menys caòtic i per tant, un sistema vial més organitzat. Per tal d'aconseguir aquesta conducció segura i ordenada, podem comptar amb una inversió de milions a nivell mundial.

Segons la *Direcció General de Trànsit*, cada dia es reposen entre 30 i 50 senyals de trànsit. La reposició d'un senyal, sense incloure el pal, ronda entre 90 i 300 euros segons la seva grandària i forma. Recordem que els senyals estan fabricats de metall, acer galvanitzat o alumini, un metall al qual es dóna la forma que es busca, segons la indicació del senyal, i unes dimensions concretes.

També hem de tenir en compte la consegüent etapa d'imprimació i protecció abans de col·locar-lo per tal de que sigui apte per estar a la intempèrie.

Per últim, enfatitzem la ineficiència que suposa aquest sistema en el moment en què es necessita efectuar una modificació dels límits establerts o bé un canvi de la senyalització, donat que és necessari manipular físicament el senyal de trànsit. Com a dada interessant, quan es va reduir a Espanya a l'any 2011 la velocitat màxima de 120 km/h a 110 km/h, va suposar un cost econòmic de 230.000€.

Per tant, podem concloure com a mínim la necessitat d'una certa telegestió, és a dir, la



Figura A.3: Agents de trànsit modificant senyals vials [25]

possibilitat de poder gestionar a distància aquesta sèrie de modificacions en tot moment segons les condicions del trànsit pertinents. El semàfor ha estat un invent totalment necessari i contemporani tant a les necessitats del passat com les actuals, però sense deixar el punt de vista en el que resulta un cost a nivell econòmic elevat.

No només hi ha un impacte econòmic, sino que es tracta d'un sistema lent ja que el temps que transcorre des que s'efectua la ordre de modificació vial fins que finalment es realitza, és alta. Si bé algunes carreteres radials com per exemple la M-30 disposen d'un panell actualitzable, l'objectiu seria poder monitoritzar qualsevol node.

La conclusió d'aquesta secció és la de plantejar l'eliminació dels senyals de trànsit donat tots els inconvenients analitzats, tals com: manteniment i cost econòmic, ambigüetat (en situacions mal dimensionades) i 'estaticitat'.

A.1.3. Inicis del semàfor

La Història del Semàfor gairebé mereix un capítol a part. El primer antecedent del semàfor, que ve inspirat dels senyals lluminosos que ja es feien servir a la xarxa ferroviària, té lloc en 1868 quan s'instal·la una mena de semàfor a l'exterior del parlament britànic de Westminster. Era obra de l'enginyer J. P. Knight, especialitzat en senyals de ferrocarril.

Tenia l'aspecte d'un senyal de ferrocarril i estava equipat amb llums vermelles i verdes que s'encenien a la nit. Va tenir un curt període de funcionament perquè va explotar matant a un policia. L'accident va descoratjar la continuació d'altres experiments fins a l'aparició de l'automòbil, i la seva multiplicació en els carrers de la ciutat va fer imprescindible reprendre el tema.

Al 1912 s'instal·la a Salt Lake City un artefacte dissenyat per un policia anomenat Lester Wire per tal d'ordenar el trànsit. A més de llums vermelles i verdes, muntades sobre suports creuats com braços, comptava amb un emissor de bronzits. Dos bronzits indicaven el pas del trànsit que venia per l'avinguda, i un bronzit permetia el pas dels que venien pel carrer 105. L'inconvenient era greu; aquest semàfor havia de ser activat manualment i això requeria de la presència d'agents a les cantonades on aquests es situaven.

És Garrett Morgan, inventor afroamericà, el qual inventa els automatismes del semàfor.



Figura A.4: Semàfor primitiu accionat manualment per un agent de trànsit [25]

William Potts és reconegut com el creador del senyal de *vermell-groc-verd* de trànsit com el coneixem avui dia. El seu semàfor, construït de fusta amb una coberta de metall, utilitzava lents de llanterna de ferrocarril. Aquest sistema es va mantenir en servei durant quatre anys fins que finalment es va instituir la interconnexió elèctrica dels senyals de trànsit, fent servir torres perquè poguéssin ser controlades per un oficial de policia des d'una única ubicació.

Comença l'exitosa i indispensable vida d'aquest codi visual de tres colors que ha perdurat durant gairebé un segle, amb relativament pocs canvis, excepte per la incorporació de tecnologies modernes com ara temporitzadors automàtics, llums de díodes i sensors de moviment. Si bé és un sistema de control que no ha patit gaires canvis, no hem d'oblidar la dificultat que suposava, i de fet suposa, controlar el trànsit aconseguint certa fluidesa i seguretat.

Com a conclusió, aquest concepte de regulador de trànsit és prou extrapolable a les comunicacions inalàmbriques com per exemple el WiFi, una tecnologia half-duplex en la qual en tot moment només es dona una circulació en un únic sentit. És a dir, no es poden enviar i rebre dades a la vegada pel simple motiu d'evitar col·lisions entre paquets.

Aquest projecte ha suposat la quebrantació d'aquest fet, donat que s'han eliminat els semàfors dels vehicles, tot seguint garantint la seguretat i augmentant la fluidesa i el *throughput* del sistema.

A.1.4. Nova urbanització

Durant la primera meitat del segle XIX, en plena expansió de la Revolució industrial, les ciutats que fins llavors continuaven tenint un urbanisme medieval es veuen col·lapsades per la instal·lació de les noves indústries i l'expansió demogràfica. Aquest fet va resultar en una gran reestructuració urbana mai concebuda fins aquell moment.

Per tal d'explicar-lo, ens basarem en l'Eixample de Barcelona i el *pla Cerdà* [26] com a exemple de nou urbanisme:

Cerdà aspirava a una ciutat que, tot i haver arrabassat el terreny al camp, s'articulés a través de carrers amples i espais verds. Sobre el pla, l'enginyer va dissenyar una quadrícula on imperava una geometria estricta de carrers paral·lels i perpendiculars que únicament trencaven les grans avingudes que travessaven la trama en diagonal.

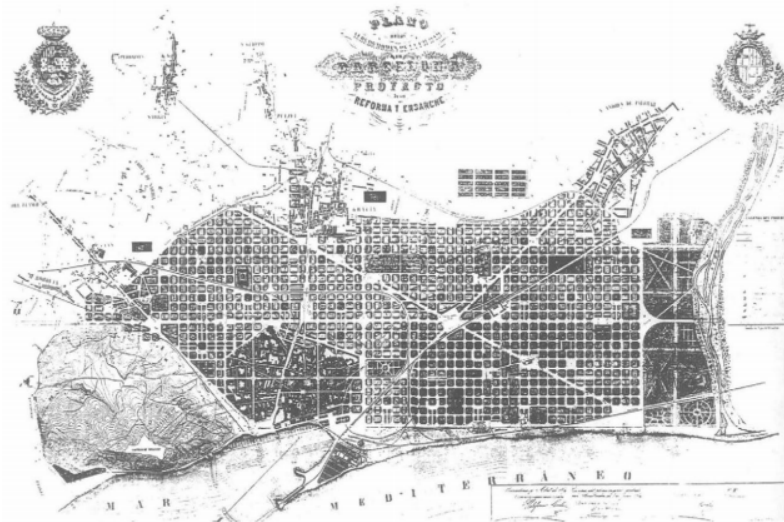


Figura A.5: Pla urbanístic de la ciutat de Barcelona [26]

La gran extensió de terreny que correspon a l'Eixample de Cerdà, és impertorbable al llarg de tot el traçat urbà i està justificada, una vegada més, en la comoditat del trànsit de persones i vehicles. Cerdà parlava de les locomotores particulars que algun dia circularien pels carrers i de la necessitat de crear un espai més ampli en cada encreuament per afavorir la circulació.

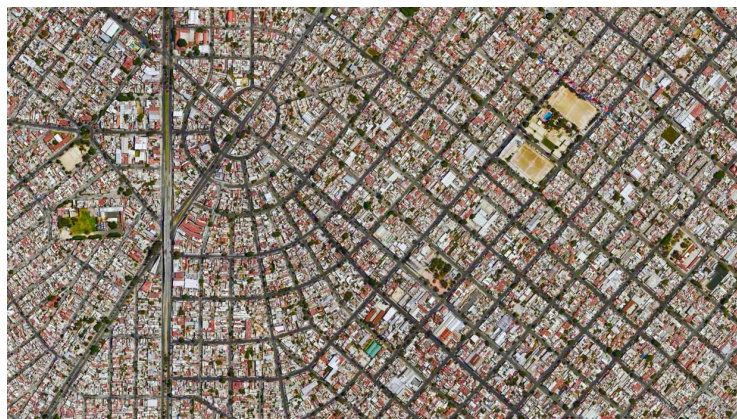


Figura A.6: Vista satèl·lit de la ciutat mexicana de Guadalajara [Wikipedia]

Aquest tipus de disseny urbà quadriculat està present a les àrees més modernes de la majoria de les grans ciutats europees així com també als Estats Units (en especial la ciutat de Los angeles), i Sud Amèrica. En especial recalquem la ciutat mexicana de Guadalajara (Figura A.6), la qual disposa d'una estructura urbana molt actualitzada degut a un accelerat creixement a nivell industrial, turístic i comercial a la dècada del 1940.

Destaquem el protagonisme que la societat li ha donat a l'automòbil donat que es tracta d'un invent que tret que era funcional amb les condicions inicials (urbanisme desorganitzat i sense asfaltar), hem tendit a adaptar el nostre urbanisme amb l'objectiu de facilitar el trànsit vehicular, augmentar la seguretat tant de vianants com conductors, i a més a més de definir un nou concepte de metròpolis.

A.2. Impacte medioambiental

El vehicle, a part de ser considerat un element clau en el funcionament de la societat moderna com hem pogut demostrar anteriorment, també esdevé una important causa de la contaminació ambiental.

El cicle vital d'un automòbil, des de la seva producció, fins al seu desballestament, és en si mateix contaminant [27]. Però el que resulta més preocupant són els nivells de contaminació atmosfèrica com a conseqüència dels gasos contaminants que s'emeten pels tubs d'escapament.

L'eliminació de contaminació és en general una tasca molt difícil i costosa, realment inviable. Per tant, les mesures preventives tenen gran importància en aquest aspecte per tal de poder reduir els nivells de contaminació. Si bé la indústria automobilística s'ha fet partícip de la sensibilitat per l'ecologia existent en la nostra societat, en part per adaptació a la legislació vigent (i també perquè l'argument de 'verd' o ecològic pot ser aprofitat com a estratègia comercial, i de fet així es fa), no és suficient.

Els principals contaminants expulsats pels automòbils són: monòxid de carboni (CO), diòxid de carboni (CO₂), òxids de nitrogen (NO_x) i hidrocarburs no cremats (HC), així com també *partícules en suspensió*. La informació de les següents subseccions han estat extretes de les fonts [28] i [29].

A.2.1. Partícules en suspensió

Les partícules en suspensió són totes les partícules sòlides i líquides que es troben suspeses en l'aire, la major part de les quals suposen un perill. Aquesta barreja complexa conté, entre altres coses, pols, pol·len, sutge, fum i petites gotes.

- **PM₁₀** (partícules gruixudes o de diàmetre aerodinàmic = 10 micres): es deu el seu origen a processos mecànics i industrials, restes vegetals i partícules primàries derivades de processos industrials entre d'altres. També té un alt grau de contaminació degut al trànsit (asfalt erosionat i restes de pneumàtics i frens generats per abrasió).
- **PM_{2,5}** (partícules fines o de diàmetre aerodinàmic = 2,5 micres) la seva composició és més tòxica, ja que els seus principals orígens són les emissions dels vehicles dièsel i aerosols orgànics secundaris.

A.2.2. Gasos nocius

- **NO_x**: és considerat el compost més nociu, perquè genera el que es coneix en l'argot científic com *smog fotoquímic*, que és la formació de components derivats que contaminen l'aire que respirem, causant problemes respiratoris i oculars. En el medi ambient l'efecte causat és la pluja àcida, que repercuteix en la vegetació per l'abradió de les zones verdes.
- **Monòxid de carboni**: aquest compost es genera per una mala combustió de gasolina, gasoil, querosè o carbó. El perill està en la seva capacitat d'acceleració de reaccions d'oxidació (òxid nítric i diòxid de nitrogen) en sòls poc oxigenats, finalment convertint-se en CO₂.
- **CO₂**: els seus efectes són devastadors perquè contribueix a elevar la temperatura mitjana de la Terra, que és de 15 graus. La contaminació per CO₂ també té conseqüències per a la salut, donat que les temperatures altes provoquen un augment dels nivells d'ozó i de contaminants de l'aire que agreugen les patologies respiratòries.
- **Hidrocarburs**: aquests compostos orgànics, són generats durant la combustió i tenen efectes negatius clars sobre la nostra salut: irritacions de mucoses, dificultats per respirar i malalties cròniques relacionades amb l'aparell respiratori. Les seves elevades concentracions a l'atmosfera contribueixen a la formació del famós smog fotoquímic. Poden produir gas metà (CH₄), que té un efecte directe sobre l'escalfament global, ja que contribueix a l'efecte hivernacle fins i tot 20 vegades més que el CO₂.
- **Nitrogen molecular**: l'N₂ és el gas que produeixen en major quantitat tant els motors dièsel com els de gasolina. Pot causar problemes hormonals o una possible relació amb processos de càncer.

A.2.3. Altres contaminants

A la llista d'emissions, podríem afegir el diòxid de sofre (SO₂), que té un efecte directe sobre la pluja àcida, encara que només el 2% el produeixen els motors dels cotxes, els compostos orgànics volàtils (COV), que són hidrocarburs gasosos i altres metalls procedents dels catalitzadors dels motors.

Per últim, comentar la contaminació deguda al plom; aquest metall es fa servir per millorar l'octanatge del combustible, però quan ens exposem a altes concentracions pot afectar-nos al sistema nerviós, el fetge i el ronyó. També pot danyar la qualitat dels cultius i té un impacte negatiu sobre els ecosistemes aquàtics degut a l'alta capacitat de retenció d'aquest metall per part de peixos i crustacis.

A.2.4. Conseqüències en la qualitat de l'aire

A títol d'exemple que aglutina com apareixen a la pràctica aquests contaminants, a la Taula A.1 es mostren els principals contaminants primaris de la ciutat de Madrid, presents en una atmosfera urbana junt amb les quantitats anuals que s'emeten de cada contaminant i el sector que més contribueix a la seva emissió.

Com es pot observar a la Taula A.1, és el trànsit vehicular el principal causant de la contaminació en una atmosfera urbana, amb una contribució superior al 75% en aquells contaminants que més preocupen en aquest tipus d'atmosferes com són les partícules materials (PM₁₀ i PM_{2,5}) i els òxids de nitrogen (NO_x).

Paràmetre	T/Any	Sector de contribució més important
SO ₂	3.159	Plantes de combustió no industrial (68,5%)
NO _x	29.337	Transport per carretera (77,0%)
PM _{2,5}	1.694	Transport per carretera (81,3%)
PM ₁₀	2.127	Transport per carretera (74,9%)
CO	94.291	Transport per carretera (91,4%)
Pb	4,94	Transport per carretera (52,9%)
CO ₂	8.352.000	Transport per carretera (51,1%)

Taula A.1: Emissions anuals dels contaminants primaris a la ciutat de Madrid, 2010 [29]

Font: Ayuntamiento de Madrid

Actualment a Barcelona i Madrid ens trobem en una situació gairebé crítica en quant als nivells de contaminació atmosfèrica degut a què els nivells de diòxid de nitrogen (NO₂) es troben per sobre dels permisos, esdevenint doncs, un important risc per a la salut dels ciutadans. Per tal de reduir aquests nivells de contaminació, s'han estat prenent mesures com ara limitar la circulació de vehicles segons la matrícula (parell o imparell), a més a més de limitar la velocitat de les vies.



Figura A.7: Vistes del 'núvol' d'alta contaminació que rodeja a Madrid

En resum, ens trobem en ciutats en les quals tenim milions de vehicles circulant degut a què en els nuclis urbans es concentra molta població. Si gran part de la població fa servir

aquest transport, augmenta la contaminació a nivells per sobre dels admesos i finalment acabem limitant el sistema per tal de reduir aquests nivells de contaminació i així tornar a una situació controlada però ineficient en termes de desplaçament.

A l'any 2015 es va fer un estudi a Valladolid sobre la variació de nivells de Nitrogen molecular (NO_2) i Ozó (O_3) respecte del dia sense cotxes (trànsit més limitat i en menys abundància) i el dia anterior, de trànsit regular.

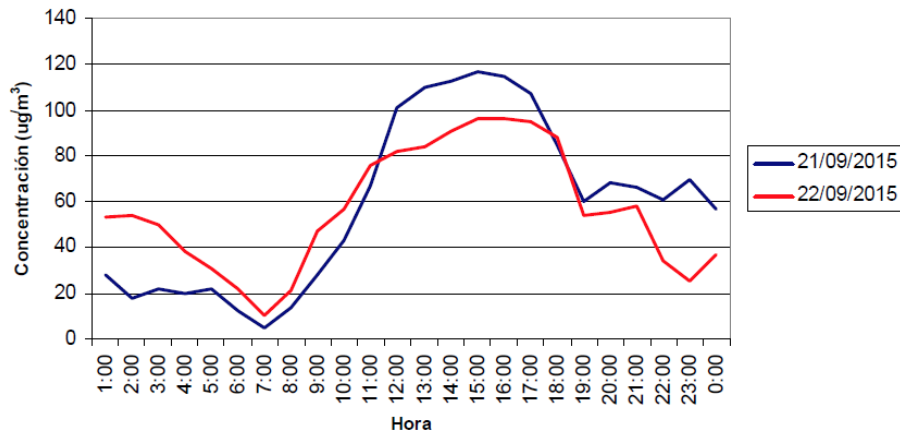


Figura A.8: Evolució de l' O_3 durant el dia sense cotxes 2015 [30]

Font: Ayuntamiento de Valladolid

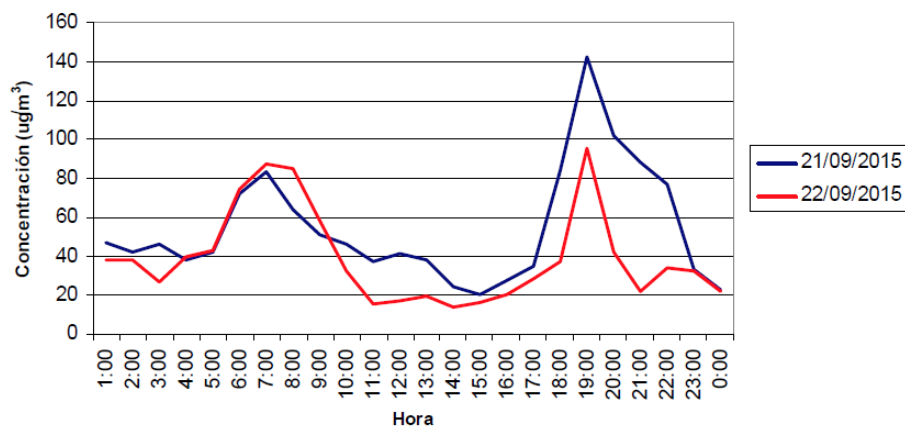


Figura A.9: Evolució del NO_2 durant el dia sense cotxes 2015 [30]

Font: Ayuntamiento de Valladolid

Com podem observar, els nivells de l' NO_2 es veuen reduïts notablement, a més a més, dels menors nivells de CO_2 , en conseqüència es veuen també reduïts els nivells d' O_3 .

És evident que el sistema de trànsit actual no és sostenible en un futur (ni de fet ho està sent actualment) ja que no permet cobrir les necessitats de la població si volem garantir una qualitat de l'aire suficientment saludable i alhora poder emprar amb llibertat el transport privat.

Els principals culpables: **els combustibles fòssils**.

Hem pogut observar que la majoria de les partícules i gasos nocius per l'atmosfera provenen de restes de combustions pertanyents de la gasolina i el dièsel, fet que ens fa plantejar la necessitat d'invertir en vehicles elèctrics i deixar en l'oblit els antics vehicles de combustió fòssil.

Per últim, un cop observades les repercussions a nivell medioambiental, el següent pas immediat és analitzar com aquests nivells de contaminació atmosfèrica posen en perill la salut humana, i en quina magnitud.

A.2.5. Mortalitat per contaminació atmosfèrica

Segons un estudi de *l'Agència Europea del Medi Ambient* (AEMA) efectuat el 2017 [31], unes 3 milions de morts estan relacionades amb la contaminació al món com a conseqüència de l'exposició a la contaminació atmosfèrica. A Espanya concretament van morir al 2013 aproximadament 29.000 a causa de contaminants existents a l'aire: 23.940 persones per mort prematura a causa de partícules contaminants de l'aire, 4.280 a causa de l'NO₂ i 1.760 més per sobreexposició a l'O₃.

Les noves dades revelen un vincle més estret entre l'exposició a la contaminació atmosfèrica en general i la de l'aire d'interiors i les malalties cardiovasculars, com els accidents cerebrovasculars, així com la contaminació atmosfèrica i el càncer.

L'Organització Mundial de la Salut (OMS) recomana que no es superin els 50 micrograms de partícules per metre cúbic d'aire de mitjana en 24 hores [32]. Oviedo va superar aquest límit el 44% dels dies; Còrdova, el 40%; Sevilla, el 23%; Madrid, el 13,5%; i Sant Sebastià, el 8%. Per cada 10 micrograms extra, el risc relatiu de mort prematura augmenta un 0,9%. No hi ha cap límit per sota del qual no s'observin danys a la salut.

Un cop més, remarcuem la imperiosa necessitat de reduir els nivells de contaminació per tal d'estar menys exposats a aquests tipus d'emissions, i per tant garantir així una major salut per als ciutadans.

A.3. Mortalitat en vies de trànsit

Segons les estadístiques publicades per la Comissió Europea (informació extreta de la font [33]), 26.000 persones van perdre la vida en accidents de trànsit a les carreteres europees en 2015. Així mateix, la CE estima que al 2014, 135.000 persones van resultar ferides greus en sinistres vials a la Unió Europea i calcula el cost social dels accidents amb víctimes (mortals i ferits) en almenys 100.000 milions d'euros.

La taxa mitjana de mortalitat en carretera a la Unió en 2015 és de 51 morts per milió d'habitants, la mateixa que en els dos anys anteriors. Els usuaris vulnerables de la via (vianants, ciclistes, etc.) representen també un alt percentatge de les 135.000 persones que, segons la Comissió, van resultar ferides per accidents de trànsit.

Per països, el nombre de víctimes mortals d'accidents de trànsit segueix variant molt d'uns

països a uns altres, encara que aquestes diferències tendeixen a disminuir. Malta (26 morts en accidents per milió d'habitants), Suècia (27), Països Baixos (28) i el Regne Unit (29) registren les taxes de mortalitat més baixes de la Unió Europea el 2015. Espanya es situa a continuació, en seté lloc, amb 36 morts per milió d'habitants, una taxa idèntica a la de 2014 i un 32% menor que la de 2010.

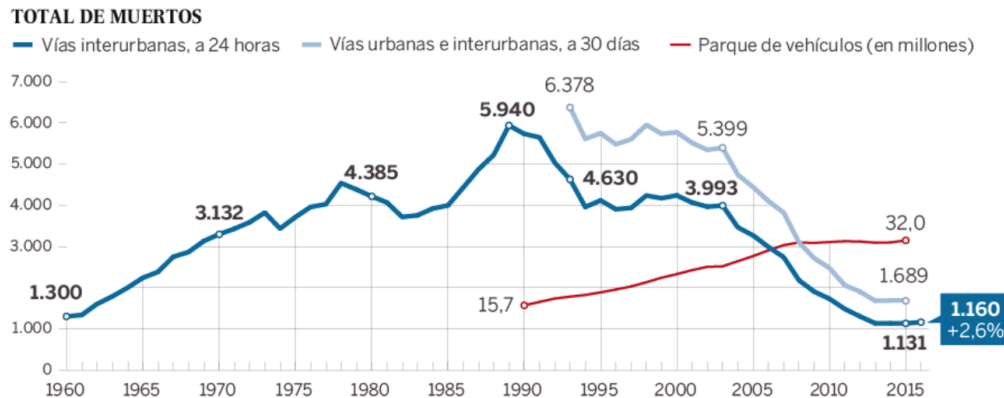


Figura A.10: Taxa de mortalitats anuals al volant a Espanya [33]

Font: www.elpais.com

Espanya concretament va acabar l'any 2015 amb unes estadístiques de 1.131 morts a vies interurbanes, i 558 morts a vies urbanes, sumant un total de 1.689 morts registrades. D'aquestes dades podem extreure les següents conclusions:

- Tot i que la mortalitat en vies de trànsit ha disminuït de manera notària, segueix havent-hi víctimes mortals.
- Els accidents de trànsit, especialment si són mortals, generen grans despeses a nivell econòmic, social i administratiu: és necessari la crida de serveis sanitaris, bombers si escau, agents de l'ordre, sovint anàlisi forense, despeses funeràries, tràmits de defunció administratius, serveis psicològics per a la família afectada, etc.

Els principals motius pels quals es produeixen accidents de trànsit són: distracció (28%), velocitat inadequada (21%), alcohol (11%), drogues il·legals (11%), no respectar la prioritat (9%) i per últim fatiga/somnolència (9%).

Per tant, podem deduir que la majoria d'accidents de trànsit són previsibles i evitables, i tot i que actualment existeixen dispositius que coneixen salvar vides i reduir la taxa de mortalitat, es fa evident la necessitat de l'existència d'algun tipus de sistema de seguretat que permeti evitar o almenys reduir aquest tipus d'accidents tenint en compte els errors humans.

Per últim, en quant als accidents de trànsit no-mortals destaquem la seva gran quantitat, tenint a més a més una tendència molt diferent a la dels accidents de caire mortal. Si bé aquests últims han mostrat una progressió gairebé del tipus decreixent, els accidents no-mortals no tenen una tendència definida i fins i tot, no es veuen reduïts amb els anys. A més a més, segons l'estadística dels últims anys, estan augmentant principalment a les vies urbanes, mostrant-se la major mortalitat a les vies interurbanes.

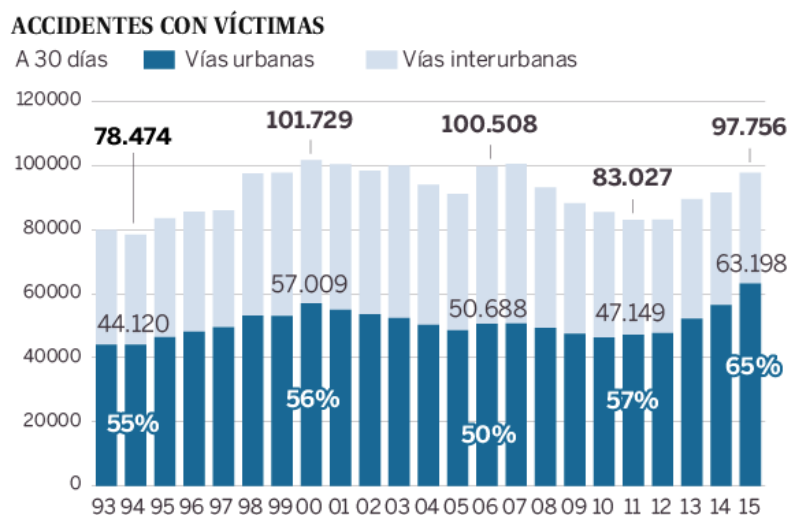


Figura A.11: Taxa d'accidents de trànsit no-mortals a Espanya [33]

Font: www.elpais.com

A.4. Ineficiències del trànsit i possibles solucions

En aquesta secció tractarem el comportament indesitjat que experimentem a les vies de trànsit. Els principals causants d'aquest fet i per últim possibles solucions i millores per al canal trànsit per tal d'augmentar la seva eficiència. La informació ha estat sintetitzada a partir del vídeo de la font [34] *Simple Solutions to Traffic*.

A.4.1. Acceleració no simultània

Un dels problemes principals de l'actual sistema de trànsit és la falta de coordinació i lentitud de reacció per part dels conductors. Plantegem la següent situació observant la Figura A.12:

Estant parats en una intersecció en la qual per exemple un semàfor es troba en vermell (1) i posteriorment canvia a verd, degut a què els cotxes comencen el seu moviment de manera discontinuada, es va acumulant un temps de retard fins que nosaltres puguem avançar (2), i podria donar-se el cas que ens quedéssim dels últims de la intersecció trobant-nos de nou el semàfor en vermell de nou (3).

Aquest fet, és un limitant del nombre de vehicles que es poden incorporar a la intersecció donat que es van acumulant més vehicles a la intersecció que volen anar a la següent, aconseguint finalment carrers molt congestionats. En general, moltes interseccions equivalen a moltes descoordinacions, i per tant, molt trànsit.

Aquest és el motiu pel qual existeixen les autopistes i autovies; no tenim interseccions, per tant no ens detenim contínuament i aleshores, no hi ha problemes de coordinació. Només es donen limitacions en aquest tipus de via a l'hora d'abandonar la via per la consegüent incorporació amb menor prioritat a una via de menor capacitat.

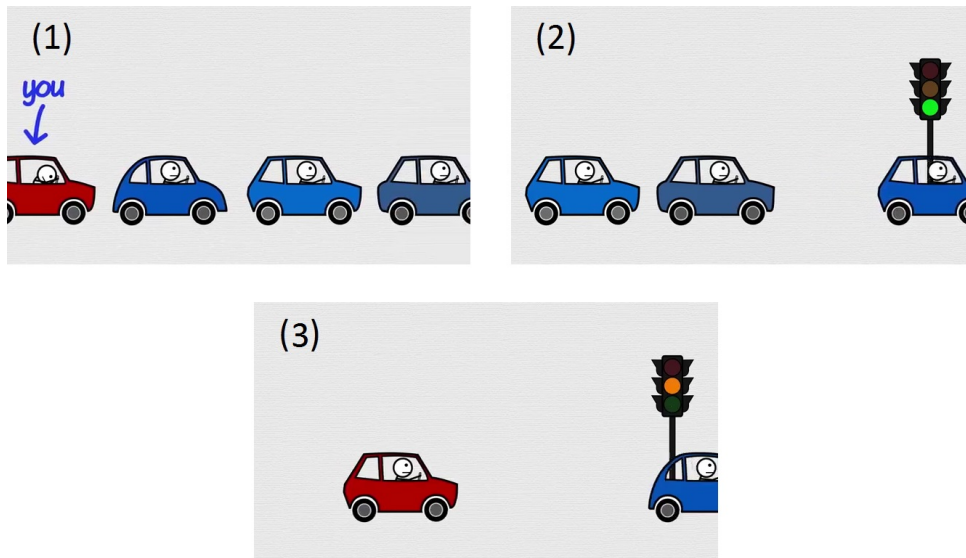


Figura A.12: Problema de descoordinació [34]

A més a més, en situacions en les que el pas a la incorporació no està regulat per una senyal vertical sino que és del tipus de cedir el pas, la decisió de quan incorporar-se dependrà de cada conductor. Lògicament es pot donar una situació en la qual un vehicle podria incorporar-se perfectament en un moment donat i el conductor decideix no fer-ho, fet que provoca que es vagi formant una cua a la incorporació.

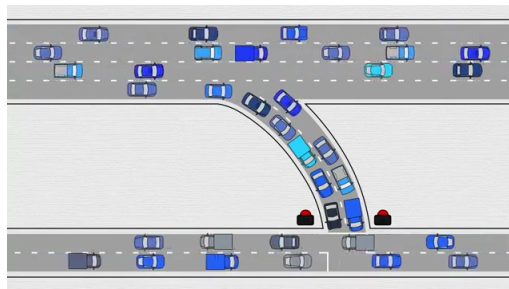


Figura A.13: Congestió al canviar de tipus de via [34]

A.4.2. Traffic snake

Aquest fenomen es pot donar en el següent escenari: ens situem en una via d'alta velocitat en la que els cotxes circulen a una velocitat més o menys constant per tal de mantenir una adequada distància de seguretat.

Observem la Figura A.14, en el moment en què un cotxe es troba un obstacle (1) i desaccelera de manera notòria, el cotxe immediatament anterior desaccelera també (2), i tot i que el primer cotxe repregui la marxa a la mateixa velocitat, aquest fet es va repetint donat que com hem comentat abans, els cotxes no reprenen la marxa de manera simultània (3) i per tant es forma el que anomenem '*Traffic Snake*' (4), esdevenint un cicle de petites retencions contínues.

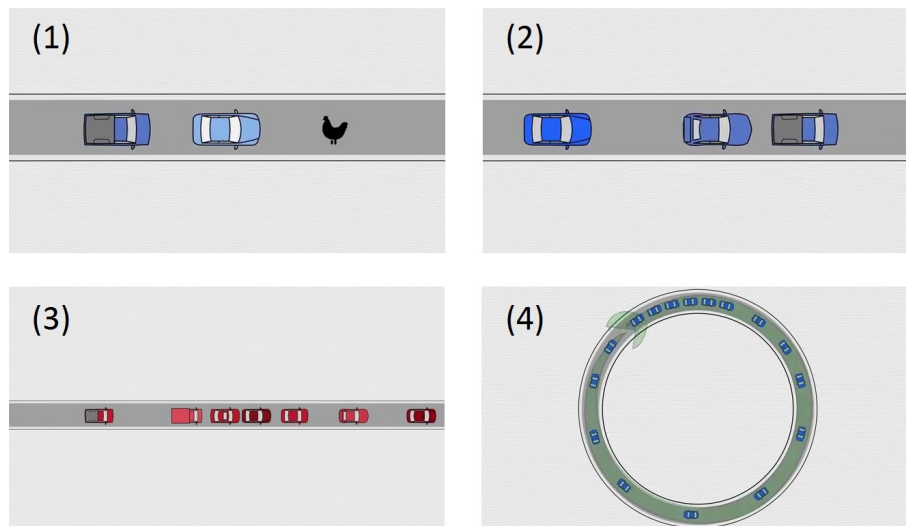


Figura A.14: Exemple de Traffic Snake [34]

El resultat final, és una intersecció 'fantasma' al mig de la via que pot acabar en una congestió temporal sense que necessàriament s'hagi produït un accident.

Així mateix, aquesta situació també es pot donar amb un simple canvi de carril mal efectuat; el cotxe del carril al qual ens estem incorporant de manera incorrecta (degut a espai insuficient per exemple), es veu obligat a frenar per evitar una possible col·lisió, i per tant, el cotxe anterior a aquest també ho farà.

Amb una conducció manual humana, tal i com estem fent actualment és evident que el millor és mantenir una certa distància de seguretat entre el vehicle posterior i l'anterior per tal de disposar de temps per reaccionar davant d'imprevistos.

Si tots els vehicles reaccionessin alhora, o almenys en un temps més reduït, no seria necessari deixar una distància de seguretat tan àmplia i aleshores la via tindria una major capacitat.

A.4.3. Platooning

Tal i com hem vist en els anteriors apartats, el fonamental problema del trànsit i a partir del qual neixen problemes derivats a aquest, és la descoordinació i limitada capacitat de reacció. Una solució a aquests inconvenients és el concepte del *platooning* o *pelotó* [35].

El platooning, és un tren de vehicles en què un vehicle guia conduït per un conductor professional, porta a la resta dels automòbils. Cada vehicle mesura la distància, la velocitat i la direcció i s'ajusta al vehicle que té davant. Tots els vehicles estan separats els uns dels altres i poden abandonar la formació en qualsevol moment. No obstant això, mentre formen part del comboi, els conductors poden relaxar-se i fer altres coses mentre el tren de carretera segueix avançant sense perill cap al seu destí a llarga distància.

Ofereix la perspectiva d'una major seguretat a la carretera, millor utilització de l'espai a la carretera, major confort del conductor especialment en viatges llargs i menor consum de

combustible degut a les reduïdes accelerades i frenades, donant lloc a menors emissions de CO₂ fins a un 20% de cara a un futur a mig termini.

El platooning està dissenyat per millorar diverses coses: en primer lloc, la seguretat a la carretera ja que redueix al mínim el factor humà, que és la causa d'almenys el 80% dels accidents de trànsit. A més a més, donat que els vehicles viatgen a velocitat d'autopista amb només uns metres de separació entre ells, es redueix per tant la distància mínima de seguretat de la que hem parlat anteriorment.

Per tant podem concloure que el platooning és una solució per alleujar la congestió del trànsit, tot i que té una aplicació més complexa en entorns urbans amb les infraestructures actuals.

A.4.4. La conducció humana

Un cop tractats tots aquests aspectes, arribem a la conclusió que el principal problema és la limitada capacitat de reacció de la que disposem els humans. Molts accidents es podrien evitar si poguéssim reaccionar a temps. És degut a aquest motiu pel qual sembla obvi que a menor velocitat, disposem de més temps de reacció i per tant, és un dels factors a tenir en compte a l'hora d'implementar una via i establir la seva velocitat de disseny.

També hem de tenir en compte que tenir humans al volant sent que equivocar-se és un fet propi en la naturalesa humana, posa en perill la vida de la resta de conductors que si bé podrien estar circulant correctament, poden perdre la vida per culpa de tercers. Tal i com hem vist a la secció anterior de mortalitat a vies de trànsit, hem pogut observar que els principals errors dels conductors són principalment: distracció, velocitat inadequada i drogues.

Aleshores l'opció que s'estava executant fins ara, era la de conscienciar els conductors a través de campanyes sobre la relació entre l'alcohol, les distraccions i el perill al volant, a més a més, tot tipus de multes i penalitzacions a tot qui estigués circulant en condicions no aptes.

L'opció que s'ha plantejat en aquest projecte és la de crear un sistema en el qual no hi hagi humans al control d'un vehicle, així doncs, donem l'entrada al concepte del vehicle autònom.

A.5. El vehicle autònom

Els vehicles autònoms són aquells que no necessiten d'una persona per poder conduir-se. Són capaços d'imitar i perfeccionar les capacitats humanes, adaptant la conducció a les circumstàncies que l'envolten, ja siguin límits de velocitat, vianants a la via, o condicions climatològiques adverses.

D'aquesta manera, és possible prescindir de la figura del conductor tal com el coneixem avui dia. La funció d'aquest es redueix a indicar-li al vehicle la direcció a la qual vol arribar.

L'origen dels vehicles autònoms és molt més remot del que es sol pensar. Des dels anys 40 s'han estat fent proves de guiatge de vehicles introduïnt en l'asfalt materials que podien ser detectats i seguits.

D'aquí s'ha passat a la detecció d'obstacles via radar en els anys 80 i en l'actualitat la tecnologia és un complex conjunt de tecnologies que inclouen el reconeixement de moviment per càmeres i els sistemes de detecció làser.

Existeixen diferents nivells en quant al concepte d'autonomia [36]:

- **Nivell 1, assistent de conducció:** els vehicles poden assistir al conductor en algunes tasques bàsiques. Això és, bàsicament, mantenir una velocitat fixa i mantenir el cotxe al carril.
- **Nivell 2, semi-autonomia:** un vehicle de nivell 2 pot funcionar només sempre que no sorgeixi cap contratemps. Si hi ha qualsevol problema, la màquina cedeix automàticament el control.
- **Nivell 3, autonomia controlada:** el vehicle pren decisions igual que en el nivell 2, però el cotxe compta amb la capacitat de decidir. Només es dona en entorns simples com una autopista.
- **Nivell 4, alt nivell d'autonomia:** el vehicle pot circular sense supervisió en àrees on aquest disposi de tota la informació per poder ser autònom.
- **Nivell 5, autonomia total:** només cal indicar la direcció a la que volem anar i oblidar-nos de supervisar cap acció. Aquest nivell d'autonomia és el que veritablement podria canviar la nostra manera d'entendre els vehicles i el transport.

A.5.1. Avantatges del vehicle autònom

- **Seguretat vial:** els vehicles autònoms reaccionarien de manera més ràpida i eficaç que un humà, ja que constantment monitoritza l'estat de la via. Per tant en poques paraules, les taxes de mortalitat baixaran en picat.
- **Adéu a la senyalització:** la senyalització era un sistema implementat per i pels humans, si tots féssim servir vehicles autònoms, podrien comunicar-se entre ells i per tant estalviariem tota senyalització possible.
- **Eficàcia vial:** aspectes com l'acceleració no simultània i la *traffic snake* desapareixerien per complet, per tant el resultat final seria una circulació fluida i sense congestions.
- **Aprofitament de la infraestructura vial:** no és necessari remodelar la actual infraestructura urbana, els vehicles s'adapten a l'entorn que els envolta, per tant podem aprofitar tot l'urbanisme actual.
- **Amigable amb el medi ambient:** la legislació europea estableix que a l'any 2020 almenys un 40% de la producció elèctrica ha de provenir de fonts renovables, i aleshores, les emissions d'un cotxe elèctric/autònom es situarien en poc més de

3Kg de CO₂/100 km, és a dir, aproximadament una cinquena part de les emissions d'un vehicle de gasolina actual.

- **Baixa contaminació acústica:** al no tenir un motor de combustió, el soroll que produeixen els vehicles elèctrics és tan mínim que fins i tot Europa obligarà a que produeixin més soroll artificial a partir del 2019 per tal d'evitar accidents amb els vianants.
- **Autonomia elèctrica:** actualment a l'any 2017 s'han registrat autonomies d'aproximadament 450km, sent més que suficient per un ús urbà.

A.5.2. Funcionament del vehicle autònom

Basant-nos en els últims models de la companyia Tesla, cada cotxe porta instal·lades vuit càmeres que ofereixen una visió de 360 graus al voltant del vehicle i vigilen una àrea de fins a 250 metres. A més a més, 12 sensors ultrasònics que són capaços de detectar objectes de tot tipus i grandària, davant, darrere o als costats.

En tercer lloc, un radar davanter ofereix dades addicionals fins i tot amb pluja forta, boira o pols. A la Figura A.15 podem observar com el vehicle percep el seu entorn, s'aprecia la monitorització constant de l'entorn, detectant cotxes, senyals, vianants, etc.



Figura A.15: Vehicle autònom de Tesla: detecció de l'entorn

Font: Tesla webpage

APÈNDIX B. PROTOCOLS D'ENCAMINAMENT DINÀMIC EN DETALL

B.0.1. RIP: Routing Information Protocol

RIP és un protocol vector distància (*Routing by rumour*) que es basa en sumar el cost total del camí i finalment comparar costos. Per a RIP, el cost és directament proporcional al nombre de salts entre nodes, per tant RIP compara la quantitat de salts que té cada camí i tria la ruta que considera més òptima en base a aquest criteri. No es calcula el que entenem per *Shortest path*, donat que la ruta més curta no necessàriament implica un menor nombre de nodes intermitjos, cada router tria la millor ruta i la comunica als seus veïns en adjacència directa.

RIP defineix tres tipus de temporitzadors: actualització (30s), desactivació (180s) i esborrat (300s). El temps d'actualització correspon al màxim temps a transcórrer entre l'enviament de missatges d'actualització dels veïns. El temps de desactivació correspon al temps màxim que pot esperar un router sense rebre actualitzacions per part dels veïns, i per últim, el temps d'esborrat implica que un cop transcorregut els 300s, s'esborren les rutes del router caigut.

Avantatges de RIP	
Simple	Protocol senzill de configurar degut a què no és necessari establir adjacències entre els routers, és a dir, aquests no estableixen una connexió prèvia per intercanviar informació sobre la topologia actual de la xarxa.
Lleuger	Computacionalment consumeix pocs recursos degut principalment a la seva simplicitat i el seu procés senzill per triar rutes a base de comparacions de costos.

Inconvenients de RIP	
Limitat	És un protocol molt limitat, interpreta la mètrica com a salts, esdevenint una visió molt ineficient de l'entorn. No té en compte l'amplada de banda dels ports ni la càrrega dels enllaços a l'hora de triar la millor ruta.
Mètrica ineficient	Aquest protocol no permet més de 15 salts, en cas d'excedir aquest valor, el paquet es descarta. Definitivament no resulta adient per enginyeria de trànsit, donat que és obvi que podem travessar més de 15 interseccions (nodes) fins a assolir el nostre destí.
Lenta convergència	En el moment en què es produeixen canvis a la topologia, la convergència és molt lenta (30s), a més a més, RIP triga 300s en esborrar rutes invàlides, motiu pel qual no és segur per al trànsit vial.

B.0.2. OSPF: Open Shortest Path First

Es tracta d'un protocol d'encaminament dinàmic del tipus *Link State Protocol*, és a dir, els routers d'una mateixa àrea s'intercanvien informació sobre la topologia de la xarxa a través dels paquets del tipus LSA (*Link State Advertisement*), de manera que després d'aquesta sincronització tots els routers tenen tota la informació topològica de la part de la xarxa corresponent.

Un cop els routers coneixen la topologia de la xarxa i saben els costos de cada enllaç, executen l'algoritme de Dijkstra per a cada una de les possibles destinacions amb el fi de determinar el camí més curt (Shortest Path).

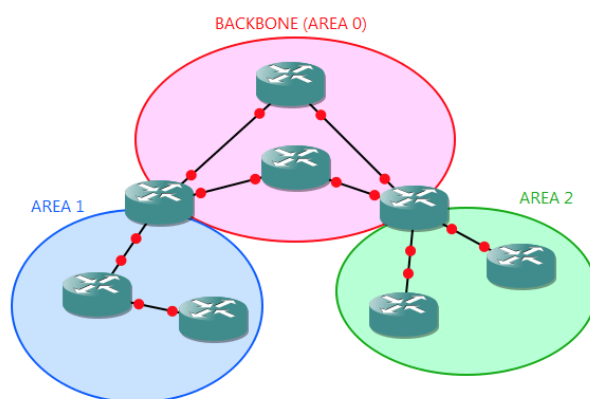


Figura B.1: Desplegament d'OSPF en àrees

OSPF defineix el concepte d'àrees de routers; entre àrees es comporta com un protocol vector-distància, només coneixent el cost, de manera que es comportaria de manera semblant a RIP en aquest cas només. Utilitza una àrea Backbone 0 que s'encarrega de distribuir informació (LSA) a les diferents àrees; es tracta d'un mecanisme que entre d'altres aspectes s'utilitza per a la prevenció de bucles, de manera que tota àrea ha d'estar connectada a l'àrea 0.

Avantatges d'OSPF	
Eficaç	Assoleix un bon control de la xarxa, cada router coneix la informació de la topologia de la xarxa, pel que permet una monitorització contínua dels enllaços.
Rutes òptimes	Les rutes que tria són sempre les millors gràcies a l'algoritme de Dijkstra i a més a més, es tenen en compte aspectes com l'amplada de banda dels enllaços entre d'altres, pel que s'està donant una visió més real de l'estat de la xarxa.
Ràpida convergència	Els canvis en l'enrutament es propaguen de manera instantània i no periòdica.

Inconvenients d'OSPF	
Complexitat	Necessita una organització adequada, donant com a resultat una difícil configuració i administració.
Cost computacional	Per tal de mantenir tota la informació de la xarxa disponible, els routers necessiten una quantitat de RAM i CPU considerable, sobretot quan es tracta d'escenaris a gran escala. Conforme augmenten els nodes, la complexitat d'executar Dijkstra esdevé exponencial.
Recàlculs continus	Cada cop que hi ha un canvi a la xarxa, es recalcula Dijkstra, s'envien LSA a l'àrea 0 i aquesta la difon a totes les àrees. S'ha de procurar que la xarxa sigui estable, donat que només un node en mal estat estaria perjudicant tota l'àrea.
Àrea backbone	L'existència d'una àrea backbone implica que totes les àrees haguessin d'estar connectades a aquesta. Aquest fet extrapolat al trànsit vial en quant al desplegament de balises no té cap sentit, donat que no es poden interconnectar àrees que no siguin la 0 entre sí.

B.0.3. IS-IS: Intermediate System to Intermediate System

IS-IS és un protocol del tipus *Link state* molt semblant a OSPF. Una diferència important entre els protocols es refereix a la forma en què operen en el model OSI. IS-IS opera en capa 2 de manera nativa (capa de dades), pel que és capaç de transmetre informació d'encaminament per a qualsevol protocol d'encaminament, no limitant-se a IP com OSPF i molts altres protocols.

Un punt a favor d'IS-IS respecte a OSPF en aquest escenari és que mentre que OSPF opera per àrees com hem comentat fins ara, IS-IS treballa amb nivells lògics, els quals ajuden al protocol a ser més escalable i a tenir una millor convergència. En general, un disseny jeràrquic redueix el consum (memòria i CPU) i organitza la topologia de xarxa tal manera que facilita la presa de decisions a més a més d'aïllar la xarxa en el moment en què un canvi pugués afectar una àrea o nivell.

Executa l'algoritme de Shortest Path Dijkstra com OSPF, de manera que tots els routers coneixen tota la topologia. La gran diferència rau a l'hora de definir el concepte d'àrea (nivells en IS-IS):

- OSPF defineix una àrea de xarxa troncal anomenada zona 0 per als anuncis inter-àrea, mentre que IS-IS categoritza el domini en dues capes.
- Qualsevol router OSPF pot pertànyer a múltiples àrees (*Area Border Router*), mentre que els routers IS-IS només poden pertànyer a una única àrea.

En referència al primer punt, IS-IS no té una àrea de xarxa troncal com l'àrea OSPF 0. L'equivalent a una àrea backbone en IS-IS és una col·lecció contigua de routers tipus L2 en adjacència amb altres routers del tipus L1/2.

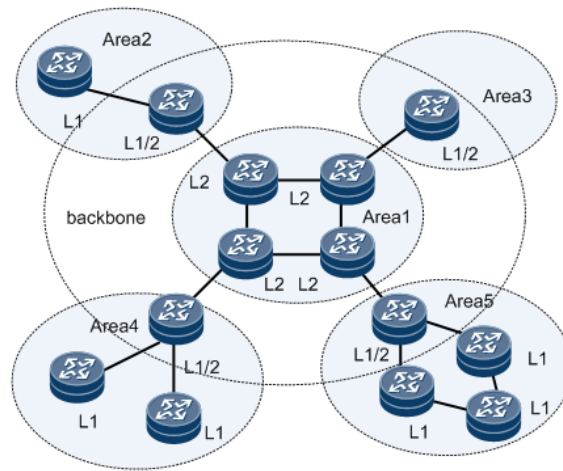


Figura B.2: Exemple de xarxes interconnectades amb IS-IS

Per tant, tot i no existir el concepte d'àrea central, es segueix fent vital la necessitat de generar una àrea amb routers L2 que es trobin en adjacència amb els routers de tipus L1/2 de la resta d'àrees (observar la Figura B.4). En resum, segueix sent prou similar a OSPF i no s'adapta a un possible desplegament a nivell de trànsit ja que per tal de transferir paquets d'una àrea a una altra, es fa necessari el pas per el router més proper de nivell L1/2 intern per tal de ser commutat al destí de diferent àrea a través d'un router de nivell L2.

Avantatges d'IS-IS	
Extensible	Molt flexible, suporta gran varietat de protocols IP i permet ser modificat a través dels paràmetres TLV (Type-Length-Value) de manera que podria carregar els atributs que nosaltres consideréssim necessaris per fer una homologia xarxa-trànsit.
Eficaç	Permet obtenir en tot moment la informació de la xarxa, pel que permet una monitorització contínua dels enllaços al ser del tipus Link-State.
Rutes òptimes	Les rutes que tria són sempre les millors gràcies a l'algorisme de Dijkstra i a més a més, es tenen en compte aspectes com l'amplada de banda dels enllaços entre d'altres.
Ràpida convergència	Els canvis en l'enrutament es propaguen de manera instantània i no periòdica.

Inconvenients d'IS-IS	
Estricte	El fet que no hi hagi una àrea 0 es resumeix en què s'ha de tenir molt controlat quines rutes es difonen entre nivells donat que és molt senzill generar bucles d'encaminament per error.
Complexe	Necessita una organització adequada, donant com a resultat una difícil configuració i administració.
Cost computacional	Els routers necessiten una quantitat de RAM i CPU considerable. Conforme augmenten els nodes, la complexitat d'executar Dijkstra esdevé exponencial.

B.0.4. BGP: Border Gateway Protocol

BGP és un protocol d'encaminament dinàmic del tipus *Path vector* (vector camí). Es basa en anunciar la ruta junt amb un conjunt d'atributs/valors per a aquesta xarxa. Aquests atributs contenen la llista de sistemes autònoms. Un sistema autònom és un conjunt de routers controlats per la mateixa entitat/organització i que té les mateixes polítiques, es tracta de routing a una escala més gran.

No està pensat per a enrutar dins d'un sistema autònom sinó entre sistemes autònoms com s'aprecia a la Figura B.3.

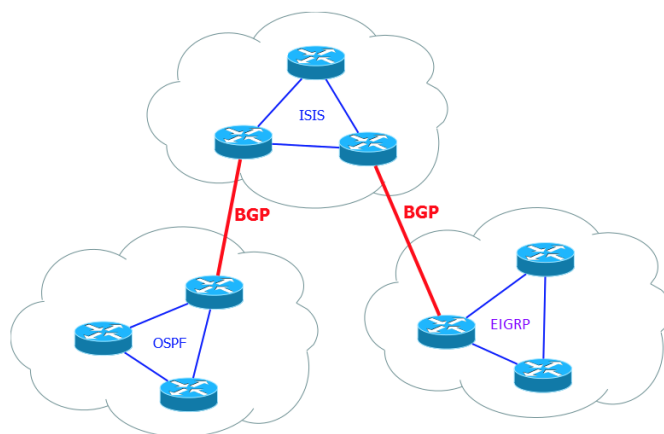


Figura B.3: Diversos sistemes autònoms amb BGP com a protocol d'enllaç

No es preocupa d'encaminar el trànsit per un camí òptim sinó que busca l'escalabilitat. Es transmeten una sèrie d'atributs, entre ells una llista de sistemes autònoms pels quals s'ha propagat la ruta. Encara que hi ha més criteris de decisió, prefereix travessar una quantitat menor de sistemes autònoms. Té mecanismes de control molt potents per tal de poder filtrar quines rutes es reben o anuncien i com seleccionar unes o altres.

A nivell d'enginyeria de trànsit té una capacitat alta sobretot en quant a escalabilitat, però no busca optimitzar l'amplada de banda com sí fan OSPF o IS-IS, sinó que es basa en polítiques i no en l'estat del canal, pel que realment no és el protocol indicat per desenvolupar aquest projecte.

Avantatges de BGP	
Escalable	BGP va ser dissenyat per fer front a xarxes de grans dimensions com Internet. És capaç de mantenir 200K-500K de direccions sense carregar a un router amb càlculs pesats ja que no es basa en el canvi dinàmic d'estats d'enllaç.
Alt control	Permet discriminar el trànsit segons la seva procedència entre d'altres, de manera que és possible establir una sèrie de normes per limitar rutes. Per exemple: evitar que el trànsit que surt o entra a IBM no transiti per Microsoft.

Inconvenients de BGP	
Convergència lenta	Els routers amb BGP no refresquen la seva taula de rutes. En lloc d'això, la informació de routing rebuda per un veí és retinguda fins que una actualització incremental és rebuda cada 30s. Si hi ha una caiguda d'un veí, sol trigar 180 segons amb la qual cosa afegeix retard, comportant-se de manera similar a RIP en aquest aspecte.
Rutes ineficients	Les rutes resulten poc òptimes des del punt de vista d'aquest projecte donat que BGP no busca el camí òptim, sino que busca la ruta amb el menor nombre total de sistemes autònoms diferents fins a accedir a la xarxa.

B.0.5. EIGRP: Enhanced Interior Gateway Routing Protocol

EIGRP és un protocol d'encaminament del tipus *Vector distància avançat*, propietat de Cisco Systems i successor de l'anterior IGRP, que ofereix el millor dels algorismes basats en Vector distància i Estat d'enllaç.

Els routers EIGRP aprenen dinàmicament noves rutes que s'uneixen a la xarxa mitjançant la formació d'adjacències i recopilacions sobre el que un router coneix sobre cada ruta. Al final del procés, quan un router aprèn una ruta d'una xarxa secundària, tot el que el router coneix és alguna mesura de la distància (la mètrica) així com el next-hop router i la conseqüent interfície de sortida a aquesta ruta (un vector, o direcció). Aquest funcionament també s'anomena *Routing by rumour*.

EIGRP utilitza un algorisme híbrid i equilibrat conegut com *Diffusing Update Algorithm* (DUAL).

Algunes de les millors funcions d'OSPF com les actualitzacions parcials i la detecció de veïns, es fan servir de manera similar amb EIGRP. Tot i que no garanteix l'ús de la millor ruta, es tracta d'un protocol molt versàtil.

Utilitza una mètrica composta per tal de determinar la millor ruta basant-se en l'amplada de banda, el retard, la confiabilitat i la càrrega de l'enllaç. El següent càlcul defineix la mètrica:

$$\text{Mètrica} = \left[K_1 BW + \frac{K_2 BW}{256 - \text{Load}_{MAX}} + K_3 \text{Delay} \right] \cdot \left[\frac{K_5}{K_4 + \text{Reliability}} \right]$$

On:

- **BW** correspon a l'amplada de banda mínima (en kilobits per segon) al llarg de la trajectòria del router a la xarxa de destinació.
- **Load** és un paràmetre que identifica la càrrega de l'enllaç en un rang entre 1 i 255, sent 255 un enllaç saturat.

- **Delay** representa la demora (en desenes de microsegons) al llarg de la trajectòria del router a la xarxa destí.
- **Reliability** fa referència a la fiabilitat d'un enllaç en un rang entre 1 i 255, sent 255 el més fiable.

Les diverses constants (K_1, K_2, \dots, K_5) es poden ajustar per l'usuari per produir comportaments variables. Un fet important és que si K_5 s'ajusta a zero, el terme $\frac{K_5}{K_4 + \text{Reliability}}$ desapareix.

El valor per defecte de K_1 i K_3 s'estableix a 1, i la resta a zero. Reduïnt la fórmula anterior a la següent expressió: $(\text{Bandwidth} + \text{Delay}) \cdot 256$.

Avantatges d'EIGRP	
Eficaç	Si bé les rutes es propaguen com un vector de distància, la mètrica es basa en l'ample de banda mínim i en el retard acumulat de la ruta en lloc del recompte de salts, motiu pel qual es classifica com un protocol vector distància avançat.
Lleuger	El fet que cada router conegui la informació a través dels seus veïns no fa necessària una gran quantitat de memòria.
Mètrica	Compta amb un bon mecanisme de càlcul de mètrica en cas de trobar una càrrega desigual, permetent als administradors de xarxa distribuir millor el fluxe del trànsit.
Ràpida convergència	EIGRP fa servir <i>Diffusing Update Algorithm</i> (DUAL), el qual permet la inserció de rutes de suport a la taula de topologia d'EIGRP. Aquestes s'utilitzen en cas de falla de la ruta principal. A causa de què es tracta d'un procediment local, el canvi a la ruta de resguard és immediat i no implica cap acció en cap altre router.
Escalable	Les actualitzacions limitades signifiquen que EIGRP utilitza un menor ample de banda, especialment en grans xarxes amb moltes rutes.
Extensible	Els canvis en l'enrutament es propaguen de manera instantània i no periòdica.

Inconvenients d'EIGRP	
Ús propietari	No es tracta d'un protocol estàndard, és utilitzat només per empreses que tenen productes Cisco.
Complexa	L'algorisme DUAL és complexa i requereix un alt cost de CPU tot i que està correctament optimitzat en escenaris a gran escala.
Rutes no-òptimes	Les rutes que conforma EIGRP no són les més òptimes degut a què aquestes són obtingudes a partir de les decisions de la resta de veïns, sense conèixer tota la topologia. Si bé és més lleuger que Dijkstra, no sempre obté les millors rutes.

APÈNDIX C. SUBNETTING EN DETALL

C.0.1. Subdivisió per barri

Considerant els següents barris de Barcelona província:



Figura C.1: Barris de Barcelona

Partim de la següent xarxa:

Xarxa base amb 16,777,214 hosts	
Address: 10.0.0.0	HostMin: 10.0.0.1
Network: 10.0.0.0/8	HostMax: 10.255.255.254
Broadcast: 10.255.255.255	Hosts/Net: 16777214

Aplicant el procés de subnetting per tal de passar de la xarxa 10.0.0.0/8 a 10.0.0.0/12, obtenim les següents subxarxes resultants:

Subxarxes dels barris de Barcelona	
Sants-Montjuïc	Les Corts
Network: 10.0.0.0/12	Network: 10.16.0.0/12
Broadcast: 10.15.255.255	Broadcast: 10.31.255.255
HostMin: 10.0.0.1	HostMin: 10.16.0.1
HostMax: 10.15.255.254	HostMax: 10.31.255.254
Hosts/Net: 1048574	Hosts/Net: 1048574

Eixample	Sarrià Sant Gervasi
Network: 10.32.0.0/12 Broadcast: 10.47.255.255 HostMin: 10.32.0.1 HostMax: 10.47.255.254 Hosts/Net: 1048574	Network: 10.48.0.0/12 Broadcast: 10.63.255.255 HostMin: 10.48.0.1 HostMax: 10.63.255.254 Hosts/Net: 1048574
Ciutat Vella	Gràcia
Network: 10.64.0.0/12 Broadcast: 10.79.255.255 HostMin: 10.64.0.1 HostMax: 10.79.255.254 Hosts/Net: 1048574	Network: 10.80.0.0/12 Broadcast: 10.95.255.255 HostMin: 10.80.0.1 HostMax: 10.95.255.254 Hosts/Net: 1048574
Horta Guinardó	Sant Martí
Network: 10.96.0.0/12 Broadcast: 10.111.255.255 HostMin: 10.96.0.1 HostMax: 10.111.255.254 Hosts/Net: 1048574	Network: 10.112.0.0/12 Broadcast: 10.127.255.255 HostMin: 10.112.0.1 HostMax: 10.127.255.254 Hosts/Net: 1048574
Nou Barris	Sant Andreu
Network: 10.128.0.0/12 Broadcast: 10.111.255.255 HostMin: 10.96.0.1 HostMax: 10.111.255.254 Hosts/Net: 1048574	Network: 10.144.0.0/12 Broadcast: 10.159.255.255 HostMin: 10.144.0.1 HostMax: 10.159.255.254 Hosts/Net: 1048574

C.0.2. Subdivisió per zona

Un cop procedit el subnetting de la xarxa base en 16 subxarxes de barri (de les quals només en faríem servir 10), subdividirem cada subxarxa de barri en 16 subxarxes menors corresponents a les zones.

L'escenari pràctic que plantejarem es basarà en els barris de Gràcia (zones 'Vila de Gràcia' i 'El camp d'En Grassot i Gràcia Nova') i Eixample (zones 'La dreta de l'Eixample' i 'Sagrada Família'). Per tant només mostrarem les subxarxes de les zones creades corresponents a aquests dos barris.

Partim de la següent xarxa:

Xarxa base amb 1,048,574 hosts	
Address: 10.32.0.0	HostMin: 10.32.0.1
Network: 10.32.0.0/12	HostMax: 10.47.255.254
Broadcast: 10.47.255.255	Hosts/Net: 1048574

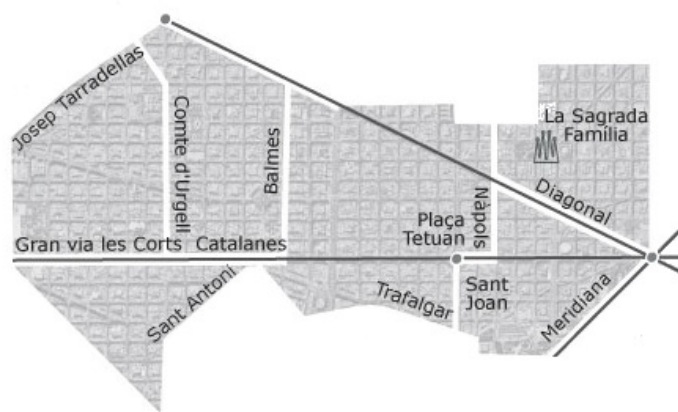


Figura C.2: Zones de l'Eixample

Aplicant el procés de subnetting per tal de passar de la xarxa 10.32.0.0/12 a 10.32.0.0/16, obtenim les següents subxarxes resultants:

Subxarxes de les zones de l'Eixample	
La nova Esquerra de l'Eixample	Sant Andreu
Network: 10.32.0.0/16 Broadcast: 10.32.255.255 HostMin: 10.32.0.1 HostMax: 10.32.255.254 Hosts/Net: 65534	Network: 10.33.0.0/16 Broadcast: 10.33.255.255 HostMin: 10.33.0.1 HostMax: 10.33.255.254 Hosts/Net: 65534
L'antiga Esquerra de l'Eixample	La dreta de l'Eixample
Network: 10.34.0.0/16 Broadcast: 10.34.255.255 HostMin: 10.34.0.1 HostMax: 10.34.255.254 Hosts/Net: 65534	Network: 10.35.0.0/16 Broadcast: 10.35.255.255 HostMin: 10.35.0.1 HostMax: 10.35.255.254 Hosts/Net: 65534
El Fort Pienc	La Sagrada Família
Network: 10.36.0.0/16 Broadcast: 10.36.255.255 HostMin: 10.36.0.1 HostMax: 10.36.255.254 Hosts/Net: 65534	Network: 10.37.0.0/16 Broadcast: 10.37.255.255 HostMin: 10.37.0.1 HostMax: 10.37.255.254 Hosts/Net: 65534

A continuació fem les subdivisions corresponents per distingir les zones del de barri de Gràcia.

Partim de la següent xarxa:



Figura C.3: Zones de Gràcia

Xarxa base amb 1,048,574 hosts	
Address: 10.80.0.0	HostMin: 10.80.0.1
Network: 10.80.0.0/12	HostMax: 10.95.255.254
Broadcast: 10.95.255.255	Hosts/Net: 1048574

Subxarxes de les zones de Gràcia	
Vallcarca i els Penitents	El Coll
Network: 10.80.0.0/16	Network: 10.81.0.0/16
Broadcast: 10.80.255.255	Broadcast: 10.81.255.255
HostMin: 10.80.0.1	HostMin: 10.81.0.1
HostMax: 10.80.255.254	HostMax: 10.81.255.254
Hosts/Net: 65534	Hosts/Net: 65534
La Salut	Vila de Gràcia
Network: 10.82.0.0/16	Network: 10.83.0.0/16
Broadcast: 10.82.255.255	Broadcast: 10.83.255.255
HostMin: 10.82.0.1	HostMin: 10.83.0.1
HostMax: 10.82.255.254	HostMax: 10.83.255.254
Hosts/Net: 65534	Hosts/Net: 65534
El camp d'En Grassot i Gràcia Nova	
Network: 10.84.0.0/16	HostMax: 10.84.255.254
Broadcast: 10.84.255.255	Hosts/Net: 65534
HostMin: 10.84.0.1	

C.0.3. Subdivisions per carrer

Finalment, segons el pla de Subnetting, per cada zona crearem 256 subxarxes.

Degut a la seva magnitud, per cada zona només s'indicarà la xarxa base des de la qual es partirà per fer el subnetting, junt amb la primera i la darrera subxarxa creades.

- **Subxarxes dels carrers de 'La dreta de l'Eixample' (barri Eixample)**

Xarxa base amb 65534 hosts	
Address: 10.35.0.0 Network: 10.35.0.0/16 Broadcast: 10.35.255.255	HostMin: 10.35.0.1 HostMax: 10.35.255.254 Hosts/Net: 65534
Subxarxes de carrers zona 'La dreta de l'Eixample'	
Primera subxarxa	Darrera subxarxa
Network: 10.35.0.0/24 Broadcast: 10.35.0.255 HostMin: 10.35.0.1 HostMax: 10.35.0.254 Hosts/Net: 254	Network: 10.35.255.0/24 Broadcast: 10.35.255.255 HostMin: 10.35.255.1 HostMax: 10.35.255.254 Hosts/Net: 254

- **Subxarxes dels carrers de 'La Sagrada Família' (barri Eixample)**

Xarxa base amb 65534 hosts	
Address: 10.37.0.0 Network: 10.37.0.0/16 Broadcast: 10.37.255.255	HostMin: 10.37.0.1 HostMax: 10.37.255.254 Hosts/Net: 65534
Subxarxes de carrers zona 'La Sagrada Família'	
Primera subxarxa	Darrera subxarxa
Network: 10.37.0.0/24 Broadcast: 10.37.0.255 HostMin: 10.37.0.1 HostMax: 10.37.0.254 Hosts/Net: 254	Network: 10.37.255.0/24 Broadcast: 10.37.255.255 HostMin: 10.37.255.1 HostMax: 10.37.255.254 Hosts/Net: 254

- **Subxarxes dels carrers de 'Vila de Gràcia' (barri de Gràcia)**

Xarxa base amb 65534 hosts	
Address: 10.83.0.0 Network: 10.83.0.0/16 Broadcast: 10.83.255.255	HostMin: 10.83.0.1 HostMax: 10.83.255.254 Hosts/Net: 65534
Subxarxes de carrers zona 'Vila de Gràcia'	
Primera subxarxa	Darrera subxarxa
Network: 10.83.0.0/24 Broadcast: 10.83.0.255 HostMin: 10.83.0.1 HostMax: 10.83.0.254 Hosts/Net: 254	Network: 10.83.255.0/24 Broadcast: 10.83.255.255 HostMin: 10.83.255.1 HostMax: 10.83.255.254 Hosts/Net: 254

- **Subxarxes dels carrers de 'En Grossot I Gràcia Nova' (barri de Gràcia)**

Xarxa base amb 65534 hosts	
Address: 10.84.0.0 Network: 10.84.0.0/16 Broadcast: 10.84.255.255	HostMin: 10.84.0.1 HostMax: 10.84.255.254 Hosts/Net: 65534
Subxarxes de carrers zona 'Camp d'E.G I Gràcia Nova'	
Primera subxarxa	Darrera subxarxa
Network: 10.84.0.0/24 Broadcast: 10.84.0.255 HostMin: 10.84.0.1 HostMax: 10.84.0.254 Hosts/Net: 254	Network: 10.84.255.0/24 Broadcast: 10.84.255.255 HostMin: 10.84.255.1 HostMax: 10.84.255.254 Hosts/Net: 254

APÈNDIX D. TAULES DE RUTES

Aquest apèndix contendrà totes les taules d'encaminament obtingudes a l'exercici elaborat amb GNS3.

```
IOU1#show ip route
```

```
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external
```

```
Gateway of last resort is not set
```

```

    10.0.0.0/8 is variably subnetted, 62 subnets, 2 masks
D       10.35.0.0/24 [90/2560000] via 10.83.6.2, 00:10:56, Ethernet0/3
D       10.35.1.0/24 [90/3072000] via 10.83.6.2, 00:10:56, Ethernet0/3
                        [90/3072000] via 10.83.0.2, 00:10:56, Ethernet0/2
D       10.35.2.0/24 [90/3584000] via 10.83.6.2, 00:10:46, Ethernet0/3
                        [90/3584000] via 10.83.0.2, 00:10:46, Ethernet0/2
D       10.35.3.0/24 [90/3072000] via 10.83.6.2, 00:04:08, Ethernet0/3
D       10.35.4.0/24 [90/3584000] via 10.83.6.2, 00:05:37, Ethernet0/3
                        [90/3584000] via 10.83.0.2, 00:05:37, Ethernet0/2
D       10.35.5.0/24 [90/4096000] via 10.83.6.2, 00:05:30, Ethernet0/3
                        [90/4096000] via 10.83.0.2, 00:05:30, Ethernet0/2
D       10.35.6.0/24 [90/3584000] via 10.83.6.2, 00:01:46, Ethernet0/3
D       10.35.7.0/24 [90/4096000] via 10.83.6.2, 00:01:46, Ethernet0/3
                        [90/4096000] via 10.83.0.2, 00:01:46, Ethernet0/2
D       10.35.8.0/24 [90/4608000] via 10.83.6.2, 00:01:35, Ethernet0/3
                        [90/4608000] via 10.83.0.2, 00:01:35, Ethernet0/2
D       10.35.9.0/24 [90/2560000] via 10.83.6.2, 00:11:08, Ethernet0/3
D       10.35.10.0/24 [90/3072000] via 10.83.6.2, 00:04:08, Ethernet0/3
D       10.35.11.0/24 [90/3072000] via 10.83.6.2, 00:10:56, Ethernet0/3
                        [90/3072000] via 10.83.0.2, 00:10:56, Ethernet0/2
D       10.35.12.0/24 [90/3584000] via 10.83.6.2, 00:05:37, Ethernet0/3
                        [90/3584000] via 10.83.0.2, 00:05:37, Ethernet0/2
D       10.35.13.0/24 [90/3584000] via 10.83.6.2, 00:10:46, Ethernet0/3
                        [90/3584000] via 10.83.0.2, 00:10:46, Ethernet0/2
D       10.35.14.0/24 [90/4096000] via 10.83.6.2, 00:05:30, Ethernet0/3
                        [90/4096000] via 10.83.0.2, 00:05:30, Ethernet0/2
```

Figura D.1: Taula de rutes del router IOU1 sense sumaritzar (part 1)

```

D      10.37.0.0/24 [90/4096000] via 10.83.6.2, 00:10:37, Ethernet0/3
                        [90/4096000] via 10.83.0.2, 00:10:37, Ethernet0/2
D      10.37.1.0/24 [90/4608000] via 10.83.6.2, 00:10:28, Ethernet0/3
                        [90/4608000] via 10.83.0.2, 00:10:28, Ethernet0/2
D      10.37.2.0/24 [90/4608000] via 10.83.6.2, 00:05:20, Ethernet0/3
                        [90/4608000] via 10.83.0.2, 00:05:20, Ethernet0/2
D      10.37.3.0/24 [90/5120000] via 10.83.6.2, 00:05:12, Ethernet0/3
                        [90/5120000] via 10.83.0.2, 00:05:12, Ethernet0/2
D      10.37.4.0/24 [90/5120000] via 10.83.6.2, 00:01:25, Ethernet0/3
                        [90/5120000] via 10.83.0.2, 00:01:25, Ethernet0/2
D      10.37.5.0/24 [90/5632000] via 10.83.6.2, 00:01:15, Ethernet0/3
                        [90/5632000] via 10.83.0.2, 00:01:15, Ethernet0/2
D      10.37.6.0/24 [90/3584000] via 10.83.6.2, 00:14:31, Ethernet0/3
                        [90/3584000] via 10.83.0.2, 00:14:31, Ethernet0/2
D      10.37.7.0/24 [90/4096000] via 10.83.6.2, 00:10:37, Ethernet0/3
                        [90/4096000] via 10.83.0.2, 00:10:37, Ethernet0/2
D      10.37.8.0/24 [90/4608000] via 10.83.6.2, 00:05:20, Ethernet0/3
                        [90/4608000] via 10.83.0.2, 00:05:20, Ethernet0/2
D      10.37.9.0/24 [90/4096000] via 10.83.6.2, 00:14:18, Ethernet0/3
                        [90/4096000] via 10.83.0.2, 00:14:18, Ethernet0/2
D      10.37.10.0/24 [90/4608000] via 10.83.6.2, 00:10:28, Ethernet0/3
                        [90/4608000] via 10.83.0.2, 00:10:28, Ethernet0/2
D      10.37.11.0/24 [90/5120000] via 10.83.6.2, 00:05:12, Ethernet0/3
                        [90/5120000] via 10.83.0.2, 00:05:12, Ethernet0/2
D      10.37.12.0/24 [90/4608000] via 10.83.6.2, 00:14:08, Ethernet0/3
                        [90/4608000] via 10.83.0.2, 00:14:08, Ethernet0/2
D      10.37.13.0/24 [90/5120000] via 10.83.6.2, 00:10:21, Ethernet0/3
                        [90/5120000] via 10.83.0.2, 00:10:21, Ethernet0/2
D      10.37.14.0/24 [90/5632000] via 10.83.6.2, 00:04:36, Ethernet0/3
                        [90/5632000] via 10.83.0.2, 00:04:36, Ethernet0/2
C      10.83.0.0/24 is directly connected, Ethernet0/2
L      10.83.0.1/32 is directly connected, Ethernet0/2
D      10.83.1.0/24 [90/1536000] via 10.83.0.2, 00:33:50, Ethernet0/2
D      10.83.2.0/24 [90/1536000] via 10.83.6.2, 00:29:35, Ethernet0/3
D      10.83.3.0/24 [90/2048000] via 10.83.6.2, 00:29:35, Ethernet0/3
                        [90/2048000] via 10.83.0.2, 00:29:35, Ethernet0/2
D      10.83.4.0/24 [90/2048000] via 10.83.6.2, 00:14:55, Ethernet0/3
D      10.83.5.0/24 [90/2560000] via 10.83.6.2, 00:14:55, Ethernet0/3
                        [90/2560000] via 10.83.0.2, 00:14:55, Ethernet0/2
C      10.83.6.0/24 is directly connected, Ethernet0/3
L      10.83.6.1/32 is directly connected, Ethernet0/3

```

Figura D.2: Taula de rutes del router IOU1 sense sumaritzar (part 2)

```

D      10.83.7.0/24 [90/1536000] via 10.83.6.2, 00:29:42, Ethernet0/3
D      10.83.8.0/24 [90/2048000] via 10.83.6.2, 00:15:04, Ethernet0/3
D      10.83.9.0/24 [90/1536000] via 10.83.0.2, 00:29:37, Ethernet0/2
D      10.83.10.0/24 [90/2048000] via 10.83.6.2, 00:29:35, Ethernet0/3
                        [90/2048000] via 10.83.0.2, 00:29:35, Ethernet0/2
D      10.83.11.0/24 [90/2560000] via 10.83.6.2, 00:14:55, Ethernet0/3
                        [90/2560000] via 10.83.0.2, 00:14:55, Ethernet0/2
D      10.83.12.0/24 [90/2048000] via 10.83.0.2, 00:29:28, Ethernet0/2
D      10.83.13.0/24 [90/2560000] via 10.83.6.2, 00:29:28, Ethernet0/3
                        [90/2560000] via 10.83.0.2, 00:29:28, Ethernet0/2
D      10.83.14.0/24 [90/3072000] via 10.83.6.2, 00:14:46, Ethernet0/3
                        [90/3072000] via 10.83.0.2, 00:14:46, Ethernet0/2
D      10.84.0.0/24 [90/2048000] via 10.83.0.2, 00:33:33, Ethernet0/2
D      10.84.1.0/24 [90/2560000] via 10.83.0.2, 00:33:19, Ethernet0/2
D      10.84.2.0/24 [90/3072000] via 10.83.0.2, 00:33:04, Ethernet0/2
D      10.84.3.0/24 [90/2560000] via 10.83.6.2, 00:29:28, Ethernet0/3
                        [90/2560000] via 10.83.0.2, 00:29:28, Ethernet0/2
D      10.84.4.0/24 [90/3072000] via 10.83.6.2, 00:29:21, Ethernet0/3
                        [90/3072000] via 10.83.0.2, 00:29:21, Ethernet0/2
D      10.84.5.0/24 [90/3584000] via 10.83.6.2, 00:29:16, Ethernet0/3
                        [90/3584000] via 10.83.0.2, 00:29:16, Ethernet0/2
D      10.84.6.0/24 [90/3072000] via 10.83.6.2, 00:14:46, Ethernet0/3
                        [90/3072000] via 10.83.0.2, 00:14:46, Ethernet0/2
D      10.84.7.0/24 [90/3584000] via 10.83.6.2, 00:14:31, Ethernet0/3
                        [90/3584000] via 10.83.0.2, 00:14:31, Ethernet0/2
D      10.84.8.0/24 [90/4096000] via 10.83.6.2, 00:14:18, Ethernet0/3
                        [90/4096000] via 10.83.0.2, 00:14:18, Ethernet0/2
D      10.84.9.0/24 [90/2560000] via 10.83.0.2, 00:29:21, Ethernet0/2
D      10.84.10.0/24 [90/3072000] via 10.83.6.2, 00:29:21, Ethernet0/3
                        [90/3072000] via 10.83.0.2, 00:29:21, Ethernet0/2
D      10.84.11.0/24 [90/3072000] via 10.83.0.2, 00:29:16, Ethernet0/2
D      10.84.12.0/24 [90/3584000] via 10.83.6.2, 00:29:16, Ethernet0/3
                        [90/3584000] via 10.83.0.2, 00:29:16, Ethernet0/2
D      10.84.13.0/24 [90/3584000] via 10.83.0.2, 00:29:09, Ethernet0/2
D      10.84.14.0/24 [90/4096000] via 10.83.6.2, 00:29:09, Ethernet0/3
                        [90/4096000] via 10.83.0.2, 00:29:09, Ethernet0/2

```

Figura D.3: Taula de rutes del router IOU1 sense sumaritzar (part 3)

```
IOU1>show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override
```

Gateway of last resort is not set

```

      10.0.0.0/8 is variably subnetted, 36 subnets, 3 masks
D      10.32.0.0/12 [90/2560000] via 10.83.6.2, 00:02:00, Ethernet0/3
D      10.37.6.0/24 [90/3584000] via 10.83.6.2, 00:16:57, Ethernet0/3
          [90/3584000] via 10.83.0.2, 00:16:57, Ethernet0/2
D      10.37.9.0/24 [90/4096000] via 10.83.6.2, 00:16:57, Ethernet0/3
          [90/4096000] via 10.83.0.2, 00:16:57, Ethernet0/2
D      10.37.12.0/24 [90/4608000] via 10.83.6.2, 00:16:57, Ethernet0/3
          [90/4608000] via 10.83.0.2, 00:16:57, Ethernet0/2
C      10.83.0.0/24 is directly connected, Ethernet0/2
L      10.83.0.1/32 is directly connected, Ethernet0/2
D      10.83.1.0/24 [90/1536000] via 10.83.0.2, 00:16:55, Ethernet0/2
D      10.83.2.0/24 [90/1536000] via 10.83.6.2, 00:17:00, Ethernet0/3
D      10.83.3.0/24 [90/2048000] via 10.83.6.2, 00:17:00, Ethernet0/3
          [90/2048000] via 10.83.0.2, 00:17:00, Ethernet0/2
D      10.83.4.0/24 [90/2048000] via 10.83.6.2, 00:16:55, Ethernet0/3
D      10.83.5.0/24 [90/2560000] via 10.83.6.2, 00:16:55, Ethernet0/3
          [90/2560000] via 10.83.0.2, 00:16:55, Ethernet0/2
C      10.83.6.0/24 is directly connected, Ethernet0/3
L      10.83.6.1/32 is directly connected, Ethernet0/3
D      10.83.7.0/24 [90/1536000] via 10.83.6.2, 00:16:55, Ethernet0/3
D      10.83.8.0/24 [90/2048000] via 10.83.6.2, 00:16:55, Ethernet0/3
D      10.83.9.0/24 [90/1536000] via 10.83.0.2, 00:17:00, Ethernet0/2
D      10.83.10.0/24 [90/2048000] via 10.83.6.2, 00:17:00, Ethernet0/3
          [90/2048000] via 10.83.0.2, 00:17:00, Ethernet0/2
D      10.83.11.0/24 [90/2560000] via 10.83.6.2, 00:16:55, Ethernet0/3
          [90/2560000] via 10.83.0.2, 00:16:55, Ethernet0/2
```

Figura D.4: Taula de rutes del router IOU1 amb sumarització per barri (part 1)

D	10.83.12.0/24	[90/2048000]	via 10.83.0.2, 00:16:57, Ethernet0/2
D	10.83.13.0/24	[90/2560000]	via 10.83.6.2, 00:16:57, Ethernet0/3
		[90/2560000]	via 10.83.0.2, 00:16:57, Ethernet0/2
D	10.83.14.0/24	[90/3072000]	via 10.83.6.2, 00:16:57, Ethernet0/3
		[90/3072000]	via 10.83.0.2, 00:16:57, Ethernet0/2
D	10.84.0.0/24	[90/2048000]	via 10.83.0.2, 00:16:55, Ethernet0/2
D	10.84.1.0/24	[90/2560000]	via 10.83.0.2, 00:16:55, Ethernet0/2
D	10.84.2.0/24	[90/4608000]	via 10.83.6.2, 00:16:57, Ethernet0/3
		[90/4608000]	via 10.83.0.2, 00:16:57, Ethernet0/2
D	10.84.3.0/24	[90/2560000]	via 10.83.6.2, 00:16:57, Ethernet0/3
		[90/2560000]	via 10.83.0.2, 00:16:57, Ethernet0/2
D	10.84.4.0/24	[90/3072000]	via 10.83.6.2, 00:16:57, Ethernet0/3
		[90/3072000]	via 10.83.0.2, 00:16:57, Ethernet0/2
D	10.84.5.0/24	[90/3584000]	via 10.83.6.2, 00:16:57, Ethernet0/3
		[90/3584000]	via 10.83.0.2, 00:16:57, Ethernet0/2
D	10.84.6.0/24	[90/3072000]	via 10.83.6.2, 00:16:57, Ethernet0/3
		[90/3072000]	via 10.83.0.2, 00:16:57, Ethernet0/2
D	10.84.7.0/24	[90/3584000]	via 10.83.6.2, 00:16:57, Ethernet0/3
		[90/3584000]	via 10.83.0.2, 00:16:57, Ethernet0/2
D	10.84.8.0/24	[90/4096000]	via 10.83.6.2, 00:16:57, Ethernet0/3
		[90/4096000]	via 10.83.0.2, 00:16:57, Ethernet0/2
D	10.84.9.0/24	[90/2560000]	via 10.83.0.2, 00:16:57, Ethernet0/2
D	10.84.10.0/24	[90/3072000]	via 10.83.6.2, 00:16:57, Ethernet0/3
		[90/3072000]	via 10.83.0.2, 00:16:57, Ethernet0/2
D	10.84.11.0/24	[90/3584000]	via 10.83.6.2, 00:16:57, Ethernet0/3
		[90/3584000]	via 10.83.0.2, 00:16:57, Ethernet0/2
D	10.84.12.0/24	[90/3584000]	via 10.83.6.2, 00:16:57, Ethernet0/3
		[90/3584000]	via 10.83.0.2, 00:16:57, Ethernet0/2
D	10.84.13.0/24	[90/4096000]	via 10.83.6.2, 00:16:57, Ethernet0/3
		[90/4096000]	via 10.83.0.2, 00:16:57, Ethernet0/2
D	10.84.14.0/24	[90/4096000]	via 10.83.6.2, 00:16:57, Ethernet0/3
		[90/4096000]	via 10.83.0.2, 00:16:57, Ethernet0/2

Figura D.5: Taula de rutes del router IOU1 amb sumarització per barri (part 2)

IOU25>show ip route

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
a - application route
+ - replicated route, % - next hop override

Gateway of last resort is not set

```
10.0.0.0/8 is variably subnetted, 36 subnets, 3 masks
D    10.35.0.0/24 [90/2048000] via 10.35.10.1, 00:17:45, Ethernet0/1
D    10.35.1.0/24 [90/2560000] via 10.35.10.1, 00:17:45, Ethernet0/1
      [90/2560000] via 10.35.6.2, 00:17:45, Ethernet0/2
D    10.35.2.0/24 [90/3072000] via 10.35.10.1, 00:17:45, Ethernet0/1
      [90/3072000] via 10.35.6.2, 00:17:45, Ethernet0/2
D    10.35.3.0/24 [90/1536000] via 10.35.10.1, 00:17:45, Ethernet0/1
D    10.35.4.0/24 [90/2048000] via 10.35.10.1, 00:17:45, Ethernet0/1
      [90/2048000] via 10.35.6.2, 00:17:45, Ethernet0/2
D    10.35.5.0/24 [90/2560000] via 10.35.10.1, 00:17:45, Ethernet0/1
      [90/2560000] via 10.35.6.2, 00:17:45, Ethernet0/2
C    10.35.6.0/24 is directly connected, Ethernet0/2
L    10.35.6.1/32 is directly connected, Ethernet0/2
D    10.35.7.0/24 [90/1536000] via 10.35.6.2, 00:17:48, Ethernet0/2
D    10.35.8.0/24 [90/2048000] via 10.35.6.2, 00:17:45, Ethernet0/2
D    10.35.9.0/24 [90/1536000] via 10.35.10.1, 00:17:48, Ethernet0/1
C    10.35.10.0/24 is directly connected, Ethernet0/1
L    10.35.10.2/32 is directly connected, Ethernet0/1
D    10.35.11.0/24 [90/2048000] via 10.35.10.1, 00:17:45, Ethernet0/1
      [90/2048000] via 10.35.6.2, 00:17:45, Ethernet0/2
D    10.35.12.0/24 [90/1536000] via 10.35.6.2, 00:17:48, Ethernet0/2
D    10.35.13.0/24 [90/2560000] via 10.35.10.1, 00:17:45, Ethernet0/1
      [90/2560000] via 10.35.6.2, 00:17:45, Ethernet0/2
D    10.35.14.0/24 [90/2048000] via 10.35.6.2, 00:17:45, Ethernet0/2
D    10.37.0.0/24 [90/3584000] via 10.35.10.1, 00:17:45, Ethernet0/1
      [90/3584000] via 10.35.6.2, 00:17:45, Ethernet0/2
D    10.37.1.0/24 [90/4096000] via 10.35.10.1, 00:17:45, Ethernet0/1
      [90/4096000] via 10.35.6.2, 00:17:45, Ethernet0/2
D    10.37.2.0/24 [90/3072000] via 10.35.10.1, 00:17:45, Ethernet0/1
      [90/3072000] via 10.35.6.2, 00:17:45, Ethernet0/2
D    10.37.3.0/24 [90/3584000] via 10.35.10.1, 00:17:43, Ethernet0/1
      [90/3584000] via 10.35.6.2, 00:17:43, Ethernet0/2
D    10.37.4.0/24 [90/2560000] via 10.35.6.2, 00:17:45, Ethernet0/2
D    10.37.5.0/24 [90/3072000] via 10.35.6.2, 00:17:45, Ethernet0/2
D    10.37.6.0/24 [90/3584000] via 10.35.10.1, 00:17:45, Ethernet0/1
      [90/3584000] via 10.35.6.2, 00:17:45, Ethernet0/2
```

Figura D.6: Taula de rutes del router IOU25 amb sumarització per barri (part 1)

```

D      10.37.7.0/24 [90/3072000] via 10.35.10.1, 00:17:45, Ethernet0/1
      [90/3072000] via 10.35.6.2, 00:17:45, Ethernet0/2
D      10.37.8.0/24 [90/2560000] via 10.35.6.2, 00:17:45, Ethernet0/2
D      10.37.9.0/24 [90/4096000] via 10.35.10.1, 00:17:45, Ethernet0/1
      [90/4096000] via 10.35.6.2, 00:17:45, Ethernet0/2
D      10.37.10.0/24 [90/3584000] via 10.35.10.1, 00:17:43, Ethernet0/1
      [90/3584000] via 10.35.6.2, 00:17:43, Ethernet0/2
D      10.37.11.0/24 [90/3072000] via 10.35.6.2, 00:17:43, Ethernet0/2
D      10.37.12.0/24 [90/4608000] via 10.35.10.1, 00:17:45, Ethernet0/1
      [90/4608000] via 10.35.6.2, 00:17:45, Ethernet0/2
D      10.37.13.0/24 [90/4096000] via 10.35.10.1, 00:17:43, Ethernet0/1
      [90/4096000] via 10.35.6.2, 00:17:43, Ethernet0/2
D      10.37.14.0/24 [90/3584000] via 10.35.6.2, 00:17:43, Ethernet0/2
D      10.80.0.0/12 [90/2560000] via 10.35.10.1, 00:08:45, Ethernet0/1
D      10.83.8.0/24 [90/2048000] via 10.35.10.1, 00:17:48, Ethernet0/1
D      10.83.11.0/24 [90/2560000] via 10.35.10.1, 00:17:45, Ethernet0/1
      [90/2560000] via 10.35.6.2, 00:17:45, Ethernet0/2
D      10.83.14.0/24 [90/3072000] via 10.35.10.1, 00:17:45, Ethernet0/1
      [90/3072000] via 10.35.6.2, 00:17:45, Ethernet0/2

```

Figura D.7: Taula de rutes del router IOU25 amb sumarització per barri (part 2)

```

IOU1>sh ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

```

Gateway of last resort is not set

```

      10.0.0.0/8 is variably subnetted, 25 subnets, 4 masks
D      10.32.0.0/12 [90/2560000] via 10.83.6.2, 00:21:50, Ethernet0/3
D      10.37.6.0/24 [90/3584000] via 10.83.6.2, 00:36:47, Ethernet0/3
          [90/3584000] via 10.83.0.2, 00:36:47, Ethernet0/2
D      10.37.9.0/24 [90/4096000] via 10.83.6.2, 00:36:47, Ethernet0/3
          [90/4096000] via 10.83.0.2, 00:36:47, Ethernet0/2
D      10.37.12.0/24 [90/4608000] via 10.83.6.2, 00:36:47, Ethernet0/3
          [90/4608000] via 10.83.0.2, 00:36:47, Ethernet0/2
C      10.83.0.0/24 is directly connected, Ethernet0/2
L      10.83.0.1/32 is directly connected, Ethernet0/2
D      10.83.1.0/24 [90/1536000] via 10.83.0.2, 00:36:45, Ethernet0/2
D      10.83.2.0/24 [90/1536000] via 10.83.6.2, 00:36:50, Ethernet0/3
D      10.83.3.0/24 [90/2048000] via 10.83.6.2, 00:36:50, Ethernet0/3
          [90/2048000] via 10.83.0.2, 00:36:50, Ethernet0/2
D      10.83.4.0/24 [90/2048000] via 10.83.6.2, 00:36:45, Ethernet0/3
D      10.83.5.0/24 [90/2560000] via 10.83.6.2, 00:36:45, Ethernet0/3
          [90/2560000] via 10.83.0.2, 00:36:45, Ethernet0/2
C      10.83.6.0/24 is directly connected, Ethernet0/3
L      10.83.6.1/32 is directly connected, Ethernet0/3
D      10.83.7.0/24 [90/1536000] via 10.83.6.2, 00:36:45, Ethernet0/3
D      10.83.8.0/24 [90/2048000] via 10.83.6.2, 00:36:45, Ethernet0/3
D      10.83.9.0/24 [90/1536000] via 10.83.0.2, 00:36:50, Ethernet0/2
D      10.83.10.0/24 [90/2048000] via 10.83.6.2, 00:36:50, Ethernet0/3
          [90/2048000] via 10.83.0.2, 00:36:50, Ethernet0/2
D      10.83.11.0/24 [90/2560000] via 10.83.6.2, 00:36:45, Ethernet0/3
          [90/2560000] via 10.83.0.2, 00:36:45, Ethernet0/2
D      10.83.12.0/24 [90/2048000] via 10.83.0.2, 00:36:47, Ethernet0/2
D      10.83.13.0/24 [90/2560000] via 10.83.6.2, 00:36:47, Ethernet0/3
          [90/2560000] via 10.83.0.2, 00:36:47, Ethernet0/2
D      10.83.14.0/24 [90/3072000] via 10.83.6.2, 00:36:47, Ethernet0/3
          [90/3072000] via 10.83.0.2, 00:36:47, Ethernet0/2
D      10.84.0.0/16 [90/2560000] via 10.83.0.2, 00:05:47, Ethernet0/2
D      10.84.0.0/24 [90/2048000] via 10.83.0.2, 00:36:45, Ethernet0/2
D      10.84.3.0/24 [90/2560000] via 10.83.6.2, 00:36:47, Ethernet0/3
          [90/2560000] via 10.83.0.2, 00:36:47, Ethernet0/2
D      10.84.6.0/24 [90/3072000] via 10.83.6.2, 00:36:47, Ethernet0/3
          [90/3072000] via 10.83.0.2, 00:36:47, Ethernet0/2

```

Figura D.8: Taula de rutes del router IOU1 amb sumariització total


```

IOU12>sh ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

```

Gateway of last resort is not set

```

      10.0.0.0/8 is variably subnetted, 25 subnets, 4 masks
D       10.32.0.0/12 [90/2560000] via 10.84.13.2, 00:23:19, Ethernet0/3
D       10.37.6.0/24 [90/3072000] via 10.84.13.2, 00:38:36, Ethernet0/3
D       10.37.9.0/24 [90/2560000] via 10.84.13.2, 00:38:36, Ethernet0/3
D       10.37.12.0/24 [90/2048000] via 10.84.13.2, 00:38:41, Ethernet0/3
D       10.83.0.0/16 [90/3072000] via 10.84.13.2, 00:08:20, Ethernet0/3
D       10.83.8.0/24 [90/5120000] via 10.84.13.2, 00:08:14, Ethernet0/3
D       10.83.11.0/24 [90/4608000] via 10.84.13.2, 00:08:14, Ethernet0/3
D       10.83.14.0/24 [90/4096000] via 10.84.13.2, 00:08:14, Ethernet0/3
D       10.84.0.0/24 [90/3072000] via 10.84.13.2, 00:38:36, Ethernet0/3
D       10.84.1.0/24 [90/3072000] via 10.84.13.2, 00:38:36, Ethernet0/3
C       10.84.2.0/24 is directly connected, Ethernet0/0
L       10.84.2.2/32 is directly connected, Ethernet0/0
D       10.84.3.0/24 [90/2560000] via 10.84.13.2, 00:38:36, Ethernet0/3
D       10.84.4.0/24 [90/2048000] via 10.84.13.2, 00:38:41, Ethernet0/3
D       10.84.5.0/24 [90/1536000] via 10.84.13.2, 00:38:41, Ethernet0/3
D       10.84.6.0/24 [90/3072000] via 10.84.13.2, 00:38:36, Ethernet0/3
D       10.84.7.0/24 [90/2560000] via 10.84.13.2, 00:38:36, Ethernet0/3
D       10.84.8.0/24 [90/2048000] via 10.84.13.2, 00:38:41, Ethernet0/3
D       10.84.9.0/24 [90/2560000] via 10.84.13.2, 00:38:36, Ethernet0/3
D       10.84.10.0/24 [90/2560000] via 10.84.13.2, 00:38:36, Ethernet0/3
D       10.84.11.0/24 [90/2048000] via 10.84.13.2, 00:38:41, Ethernet0/3
D       10.84.12.0/24 [90/2048000] via 10.84.13.2, 00:38:41, Ethernet0/3
C       10.84.13.0/24 is directly connected, Ethernet0/3
L       10.84.13.1/32 is directly connected, Ethernet0/3
D       10.84.14.0/24 [90/1536000] via 10.84.13.2, 00:38:41, Ethernet0/3

```

Figura D.9: Taula de rutes del router IOU12 amb sumarització total

IOU25>show ip route

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
a - application route
+ - replicated route, % - next hop override

Gateway of last resort is not set

```
10.0.0.0/8 is variably subnetted, 25 subnets, 4 masks
D    10.35.0.0/24 [90/2048000] via 10.35.10.1, 00:37:51, Ethernet0/1
D    10.35.1.0/24 [90/2560000] via 10.35.10.1, 00:37:51, Ethernet0/1
      [90/2560000] via 10.35.6.2, 00:37:51, Ethernet0/2
D    10.35.2.0/24 [90/3072000] via 10.35.10.1, 00:37:51, Ethernet0/1
      [90/3072000] via 10.35.6.2, 00:37:51, Ethernet0/2
D    10.35.3.0/24 [90/1536000] via 10.35.10.1, 00:37:51, Ethernet0/1
D    10.35.4.0/24 [90/2048000] via 10.35.10.1, 00:37:51, Ethernet0/1
      [90/2048000] via 10.35.6.2, 00:37:51, Ethernet0/2
D    10.35.5.0/24 [90/2560000] via 10.35.10.1, 00:37:51, Ethernet0/1
      [90/2560000] via 10.35.6.2, 00:37:51, Ethernet0/2
C    10.35.6.0/24 is directly connected, Ethernet0/2
L    10.35.6.1/32 is directly connected, Ethernet0/2
D    10.35.7.0/24 [90/1536000] via 10.35.6.2, 00:37:54, Ethernet0/2
D    10.35.8.0/24 [90/2048000] via 10.35.6.2, 00:37:51, Ethernet0/2
D    10.35.9.0/24 [90/1536000] via 10.35.10.1, 00:37:54, Ethernet0/1
C    10.35.10.0/24 is directly connected, Ethernet0/1
L    10.35.10.2/32 is directly connected, Ethernet0/1
D    10.35.11.0/24 [90/2048000] via 10.35.10.1, 00:37:51, Ethernet0/1
      [90/2048000] via 10.35.6.2, 00:37:51, Ethernet0/2
D    10.35.12.0/24 [90/1536000] via 10.35.6.2, 00:37:54, Ethernet0/2
D    10.35.13.0/24 [90/2560000] via 10.35.10.1, 00:37:51, Ethernet0/1
      [90/2560000] via 10.35.6.2, 00:37:51, Ethernet0/2
D    10.35.14.0/24 [90/2048000] via 10.35.6.2, 00:37:51, Ethernet0/2
D    10.37.0.0/16 [90/2560000] via 10.35.6.2, 00:04:02, Ethernet0/2
D    10.37.6.0/24 [90/4096000] via 10.35.10.1, 00:04:02, Ethernet0/1
      [90/4096000] via 10.35.6.2, 00:04:02, Ethernet0/2
D    10.37.9.0/24 [90/4608000] via 10.35.10.1, 00:04:02, Ethernet0/1
      [90/4608000] via 10.35.6.2, 00:04:02, Ethernet0/2
D    10.37.12.0/24 [90/5120000] via 10.35.10.1, 00:04:02, Ethernet0/1
      [90/5120000] via 10.35.6.2, 00:04:02, Ethernet0/2
D    10.80.0.0/12 [90/2560000] via 10.35.10.1, 00:28:51, Ethernet0/1
D    10.83.8.0/24 [90/2048000] via 10.35.10.1, 00:37:54, Ethernet0/1
D    10.83.11.0/24 [90/2560000] via 10.35.10.1, 00:37:51, Ethernet0/1
      [90/2560000] via 10.35.6.2, 00:37:51, Ethernet0/2
D    10.83.14.0/24 [90/3072000] via 10.35.10.1, 00:37:51, Ethernet0/1
      [90/3072000] via 10.35.6.2, 00:37:51, Ethernet0/2
```

Figura D.10: Taula de rutes del router IOU25 amb sumariització total

IOU36>show ip route

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
a - application route
+ - replicated route, % - next hop override

Gateway of last resort is not set

```
10.0.0.0/8 is variably subnetted, 25 subnets, 4 masks
D    10.35.0.0/16 [90/2560000] via 10.37.5.1, 00:06:44, Ethernet0/0
D    10.35.2.0/24 [90/3072000] via 10.37.14.1, 00:39:51, Ethernet0/1
      [90/3072000] via 10.37.5.1, 00:39:51, Ethernet0/0
D    10.35.5.0/24 [90/2560000] via 10.37.14.1, 00:39:51, Ethernet0/1
      [90/2560000] via 10.37.5.1, 00:39:51, Ethernet0/0
D    10.35.8.0/24 [90/2048000] via 10.37.5.1, 00:39:49, Ethernet0/0
D    10.37.0.0/24 [90/2560000] via 10.37.14.1, 00:39:51, Ethernet0/1
      [90/2560000] via 10.37.5.1, 00:39:51, Ethernet0/0
D    10.37.1.0/24 [90/2048000] via 10.37.14.1, 00:39:53, Ethernet0/1
D    10.37.2.0/24 [90/2048000] via 10.37.14.1, 00:39:54, Ethernet0/1
      [90/2048000] via 10.37.5.1, 00:39:54, Ethernet0/0
D    10.37.3.0/24 [90/1536000] via 10.37.14.1, 00:39:54, Ethernet0/1
D    10.37.4.0/24 [90/1536000] via 10.37.5.1, 00:39:49, Ethernet0/0
C    10.37.5.0/24 is directly connected, Ethernet0/0
L    10.37.5.2/32 is directly connected, Ethernet0/0
D    10.37.6.0/24 [90/3072000] via 10.37.14.1, 00:39:51, Ethernet0/1
      [90/3072000] via 10.37.5.1, 00:39:51, Ethernet0/0
D    10.37.7.0/24 [90/2560000] via 10.37.14.1, 00:39:51, Ethernet0/1
      [90/2560000] via 10.37.5.1, 00:39:51, Ethernet0/0
D    10.37.8.0/24 [90/2048000] via 10.37.5.1, 00:39:51, Ethernet0/0
D    10.37.9.0/24 [90/2560000] via 10.37.14.1, 00:39:51, Ethernet0/1
      [90/2560000] via 10.37.5.1, 00:39:51, Ethernet0/0
D    10.37.10.0/24 [90/2048000] via 10.37.14.1, 00:39:54, Ethernet0/1
      [90/2048000] via 10.37.5.1, 00:39:54, Ethernet0/0
D    10.37.11.0/24 [90/1536000] via 10.37.5.1, 00:39:54, Ethernet0/0
D    10.37.12.0/24 [90/2048000] via 10.37.14.1, 00:39:49, Ethernet0/1
D    10.37.13.0/24 [90/1536000] via 10.37.14.1, 00:39:49, Ethernet0/1
C    10.37.14.0/24 is directly connected, Ethernet0/1
L    10.37.14.2/32 is directly connected, Ethernet0/1
D    10.80.0.0/12 [90/2560000] via 10.37.14.1, 00:30:40, Ethernet0/1
D    10.83.8.0/24 [90/4608000] via 10.37.14.1, 00:39:50, Ethernet0/1
      [90/4608000] via 10.37.5.1, 00:39:50, Ethernet0/0
D    10.83.11.0/24 [90/4096000] via 10.37.14.1, 00:39:50, Ethernet0/1
      [90/4096000] via 10.37.5.1, 00:39:50, Ethernet0/0
D    10.83.14.0/24 [90/3584000] via 10.37.14.1, 00:39:51, Ethernet0/1
      [90/3584000] via 10.37.5.1, 00:39:51, Ethernet0/0
```

Figura D.11: Taula de rutes del router IOU36 amb sumariització total

APÈNDIX E. FUNCIONS I FITXERS DE L'APLICACIÓ ASTAR

Fitxer *InfoVelocity.txt*

```
Vel. line 1 : 50
Vel. line 2 : 20
Vel. line 3 : 40
MIN Vel. line : 20
MAX Vel. line : 50
```

Fitxer *Nodes.txt*

1	A	1	275	75
2	B	1	250	275
3	C	1	80	275
4	D	1	150	470
5	E	1	250	435
6	F	1	250	670
7	G	1	260	800
8	H	1	310	970
9	I	1	425	140
10	J	1	415	280
11	K	1	420	425
12	L	1	400	620
13	M	1	400	730
14	N	1	425	900
15	O	1	620	220
16	P	1	630	300
17	Q	1	600	450
18	R	1	600	675
19	S	1	590	810
20	J2	2	415	280
21	K2	2	420	425
22	M2	2	400	730
23	R2	2	600	675
24	F3	3	250	670
25	L3	3	400	620
26	Q3	3	600	450

Els fitxers *Connections.txt*, *Time.txt* i *Saturation.txt* no s'han inclòs degut a la seva magnitud (matrius de 26x26).

Fitxer *SearchAlgorithm.py* (conté les funcions principals)

```
from BCNMap import *
import math
import heapq
import time
```

```
class Node:
```

```
    # Constructor de la classe Node
```

```
    def __init__(self, node, father):
```

```
        self.node = node # informacio del node
```

```
        self.g = 0 # cost real del cami
```

```
        self.h = 0 # valor real de la heuristica
```

```
        self.f = 0 # suma de la heuristica i del cost real (f=g+h)
```

```
        self.time = 0 # temps total des de node origen a desti
```

```
        self.numStops = 0 # nombre de nodes recorreguts
```

```
        self.walk = 0 # distacia total des de l'origen fins al desti
```

```
    if father == None:
```

```
        self.parentsID = [] # creacio variable dels pares del node
```

```
    else:
```

```
        self.parentsID = [father.node.id]
```

```
        # cami total des de l'origen fins a aquest node:
```

```
        self.parentsID.extend(father.parentsID)
```

```
        self.g = father.g #cost acumulatiu de tots els pares del node
```

```
        self.time = father.time
```

```
        self.walk = father.walk
```

```
    self.father = father # apuntador del node al node pare
```

```
    def setEvaluation(self):
```

```
        """
```

```
        setEvaluation: calcula la funcio d'avaluacio
```

```
        """
```

```
        self.f = 0
```

```
        self.f = self.g + self.h
```

```
    def setHeuristic(self, typePreference, node_destination, city):
```

```
        """
```

```
        setHeuristic: calcula el cost aproximat i optimista que  
hi ha del node origen fins al node de desti
```

```
        """
```

```
    if typePreference == 0:
```

```
        self.h = 0
```

```

elif typePreference == 1:
    self.h = euclidianDist([self.node.x, self.node.y], \
        [node_destination.node.x, node_destination.node.y]) \
        / city.max_velocity
    if node_destination.node.line != self.node.line:

```

```

        self.h += 0

```

```

elif typePreference == 2:
    self.h = euclidianDist([self.node.x, self.node.y], \
        [node_destination.node.x, node_destination.node.y])
    self.h = 0

```

```

def setRealCost(self, costTable):

```

```

    """

```

```

    setRealCost:    fixa el cost dels nodes a partir del costTable
    que calcularem a la funcio setCostTable

```

```

    """

```

```

    j = 0
    self.g = 0

```

```

    listaPadres = self.parentsID[:: -1]
    listaPadres.append(self.node.id)

```

```

    for i in listaPadres[:: -1]:
        self.g = self.g + costTable[listaPadres[j + 1]][listaPadres[j]]
        j = j + 1

```

```

# La distancia eucladiana i la distancia Manhattan son
algoritmes que complementen a la heuristica, fent
basicament un calcul aproximat amb les coordenades

```

```

def euclidianDist(xy1, xy2):
    x = (xy1[0] - xy2[0]) ** 2
    y = (xy1[1] - xy2[1]) ** 2
    return math.sqrt(x + y)

```

```

def Expand(fatherNode, allNodes, typePreference, \
    node_destination, costTable, city):

```

```

    """

```

```

    Expand: s'expandeix recorrent els nodes veïns amb
menor cost funcional

```

```

        :returns
            - childrenList: llista de nodes expandits

    """
    llista = []

    for element in fatherNode.node.destinationDic:
        node = Node(allNodes[element - 1], fatherNode)
        llista.append(node)
        node.setHeuristic(typePreference, node_destination, city)
        node.setRealCost(costTable)
        node.setEvaluation()
        node.walk = fatherNode.walk
        node.numStops = fatherNode.numStops
        node.time = node.time + \
            fatherNode.node.destinationDic[node.node.id]

        if fatherNode.node.line == node.node.line:
            velocidad=(city.velocity_lines[fatherNode.node.line - 1] \
                / float(3.6)
            node.walk += fatherNode.node.destinationDic[node.node.id] \
                * velocidad

        node.numStops += 1

    return llista

```

```

def RemoveCycles(childrenList):

```

```

    """

```

```

        RemoveCycles: elimina aquells nodes fills que tenen
        un camí amb cicles.

```

```

        :returns

```

```

            - listWithoutCycles: retorna una llista sense cicles

```

```

    """

```

```

    listWithoutCycles = []

```

```

    for node in childrenList:

```

```

        listWithoutCycles.append(node)

```

```

        if node.node.id in node.father.parentsID:

```

```

            listWithoutCycles.remove(node)

```

```

    return listWithoutCycles

```

```

def RemoveRedundantPaths(childrenList, nodeList, partialCostTable):

```


"""

RemoveRedundantPaths: elimina els camins redundants no optims.

:params

- childrenList: llista de nodes fills a estudiar en cas que continguin camins redundants*
- nodeList : llista dels nodes a visitar*
- partialCostTable: diccionari del minim g per obtenir cada node des del node origen*

:returns

- childrenList: llista de nodes fills sense cami redundant*
- nodeList: llista de nodes a visitar sense camins redundants*
- partialCostTable: diccionari del minim g per obtenir cada node des del node origen*

"""

```
llista = []
```

```
for element in childrenList:
```

```
    llista.append(element)
```

```
    if element.node.id in partialCostTable.keys():
```

```
        if partialCostTable[element.node.id] > element.g:
            partialCostTable[element.node.id] = element.g
```

```
            for i in nodeList:
```

```
                if i.node.id == element.node.id:
                    nodeList.remove(i)
```

```
            else:
```

```
                llista.remove(element)
```

```
        else:
```

```
            partialCostTable[element.node.id] = element.g
```

```
return llista, nodeList, partialCostTable
```

```
def sorted_insertion(allNodes, childrenList):
```

""" *Sorted_insertion: a cada iteracio de l'Astar es fa un Expand dels nodes, obviamet aquests nodes apareixen desordenats i aqui s'ordenen de menor cost a major*

:returns

- nodeList: llista ordenada dels nodes expandits*

"""

```

for element in childrenList:
    i = 0

    while i < len(nodeList) and element.f > nodeList[i].f:
        i += 1

    nodeList.insert(i, element)

return nodeList

```

```

def setCostTable(typePreference, allNodes, city):
    """
    setCostTable : cost real del trajecte

    :return:
        - costTable: es un diccionari que relaciona cada node amb
          la seva adjacencia i el seu cost segons l'heuristica triada
    """
    costTable = {}
    alTiempo=0
    alDistancia=0
    alSaturacio=0
    tiempo = readCostTable("../CityInformation/bcn/Time.txt")
    saturacio = readCostTable("../CityInformation/bcn/Saturation.txt")

    if typePreference == 0:
        costTable = city.adjacency

    elif typePreference == 1:

        for i in allNodes:
            aux = {}
            for j in tiempo[i.id]:
                alSaturacio=0
                line = allNodes[j - 1].line
                alTiempo=tiempo[int(i.id)][int(j)]*1.00

                if line == allNodes[i.id - 1].line:
                    tiempo[i.id][j] = \
                        (tiempo[i.id][j] * city.velocity_lines[line - 1]) \
                        / float(3.6)
                else:
                    tiempo[i.id][j] = 0
                alDistancia = tiempo[int(i.id)][int(j)] * 0.00

            for k in saturacio:
                for l in saturacio[k]:

```

```

        if l==i.id and k==j:

            alSaturacio=999
            aux.update({int(j): alTiempo+alSaturacio})
            costTable.update({int(i.id): aux})

elif typePreference == 2:

    for i in allNodes:
        aux = {}
        for j in tiempo[i.id]:
            alSaturacio=0
            line = allNodes[j - 1].line
            alTiempo=tiempo[int(i.id)][int(j)]*0.00

            if line == allNodes[i.id - 1].line:
                tiempo[i.id][j] = \
                    (tiempo[i.id][j] * city.velocity_lines[line - 1]) \
                    / float(3.6)
            else:
                tiempo[i.id][j] = 0
            alDistancia = tiempo[int(i.id)][int(j)] * 1.00

        for k in saturacio:
            for l in saturacio[k]:

                if l==i.id and k==j:

                    alSaturacio=999
                    aux.update({int(j): alDistancia+alSaturacio})
                    costTable.update({int(i.id): aux})

    return costTable

```

```

def coord2node(coord, allNodes):
    """
    coord2node : troba el node mes proper a partir de coordenades.

    :return:
        - possible_origins: llista dels index de tots els
          nodes que corresponen al node mes proper
    """
    possible_origins = []
    aux = []

```

```

for node in allNodes:
    distanciaActual = abs(coord[0] - node.x) + \
    abs(coord[1] - node.y)
    aux.append([node.id, distanciaActual])

aux = sorted(aux, key=lambda x: x[1])

for element in aux:
    inici = aux[0][1]
    if element[1] == inici:
        possible_origins.append(element[0] - 1)

return possible_origins

def AstarAlgorithm(allNodes, coord_origin, coord_destination, \
    typePreference, city, flag_redundants):
    """
    AstarAlgorithm: funcio principal. Es la connexio entre GUI i l'AStar.
    INPUTS:
        - allNodes: llista de tots els nodes (id, nom, destiDic,
            tipus velocitat, x, y)
        - coord_origin: parella de coordenades d'origen
        - coord_destination: parella de coordenades desti
        - typePreference: enter que indica l'heuristica seleccionada:
            1 - minim Timps
            2 - minima Distancia

        - city: informacio de la ciutat
        - flag_redundants: [0/1]. indica si l'algoritme ha de remoure
            algun camí redundant (1) o no (0)

    OUTPUTS:
        - time: temps total del recorregut
        - distance: distancia total del recorregut
        - numStops: nombre total de nodes que conformen la ruta
        - num_expanded_nodes: nombre total de nodes expandits
        - depth: nombre total de nodes que conformen la ruta
            inclosos nodes 'ficticis' (J2, K2, M2...)
        - visitedNodes: llista d'IDs dels nodes recorreguts
        - idsOptimalPath: llista dels IDs dels nodes que conformen
            l'optimal path (origen-desti)
        - min_distance_origin: distancia des de les coord origen
            fins el node mes proper
        - min_distance_destination: distancia des de les coord desti
            fins el node mes proper
    """

```

```

totalTime = 0
typePreference = int(typePreference)
origenNodeList = coord2node(coord_origin , allNodes)
destinationNodeList = coord2node(coord_destination , allNodes)
visitedNodes = []
num_expanded_nodes = 0
idsOptimalPath = []
partialCostTable = {}
nodeList = []
tic = time.clock()

for id in origenNodeList:
    origen = Node(allNodes[id] , None)
    nodeList.append(origen)
    destination = Node(allNodes[destinationNodeList[0]] , None)
    min_distance_origen = manhattanDist(coord_origin , \
(nodeList[0].node.x,
nodeList[0].node.y))
    min_distande_destination = manhattanDist(coord_destination , \
(destination.node.x, destination.node.y))
    costTable = setCostTable(typePreference , allNodes , city)
    print_connections(costTable)
    num_expanded_nodes = 0
    while len(nodeList) > 0 and nodeList[0].node.name != \
destination.node.name:

        node = nodeList.pop(0)
        childrenList = Expand(node, allNodes , typePreference , \
destination , costTable , city)

        childrenList = RemoveCycles(childrenList)

        if flag_redundants == 1:
            childrenList , nodeList , partialCostTable = \
RemoveRedundantPaths(childrenList , nodeList , partialCostTable)
        else:
            totalTime += node.time
            nodeList = sorted_insertion(nodeList , childrenList)
            visitedNodes.append(node.node.id)

            num_expanded_nodes += 1
node = nodeList[0]

for i in range(len(node.parentsID) - 1, -1, -1):
    idsOptimalPath.append(node.parentsID[i])
idsOptimalPath.append(node.node.id)
depth = len(node.parentsID)
toc = time.clock()

```

```
elapsed = toc - tic
print "\n\nTemps_d'execucio:_"
print elapsed
return (node.time, node.walk, 0, node.numStops, num_expanded_nodes,
        depth, visitedNodes, idsOptimalPath, min_distance_origen,
        min_distande_destination)
```